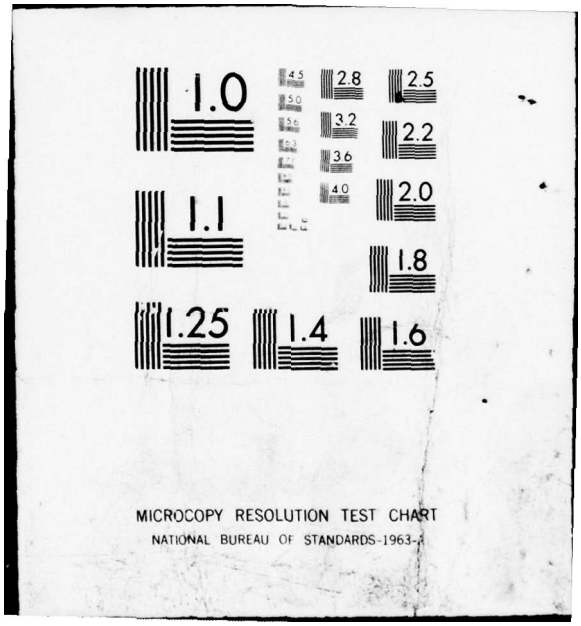
AD-A058 321 HOFFMAN MARITIME CONSULTANTS INC GLEN HEAD NY F/G 13/12
SIMULATION OF BRIDGE COLLISION INCIDENTS INVOLVING INLAND WATER--ETC(U)
JUN 78 G L PETRIE DOT-CG-72402-A
UNCLASSIFIED HMC-7764C USCG-D-49-78 NL

1 OF 1
AD A058321

END
DATE FILMED
10-78
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Report No. CG-D-49-78

(12) LEVEL II

SIMULATION OF BRIDGE COLLISION INCIDENTS INVOLVING INLAND WATERWAY TOWS: PROGRAM USERS DOCUMENTATION

G. L. PETRIE

FINAL REPORT
JUNE 1978

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
United States Coast Guard
Office of Research and Development
Washington, D.C. 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

The contents of this report do not necessarily reflect the official view or policy of the Coast Guard; and they do not constitute a standard, specification, or regulation.

This report, or portions thereof may not be used for advertising or sales promotion purposes. Citation of trade names and manufacturers does not constitute endorsement or approval of such products.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
USCG-D-49-78		

4. Title and Subtitle
Simulation of Bridge Collision Incident Involving Inland Waterway Tows: Program Users Documentation

5. Report Date
June 1978

6. Performing Organization Code

7. Author(s)
George L. Petrie

8. Performing Organization Report No.
HMC-7764C

9. Performing Organization Name and Address
Hoffman Maritime Consultants, Inc.
9 Glen Head Road
Glen Head, New York 11545

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
DOT-CG-72402-A

12. Sponsoring Agency Name and Address
Department of Transportation
U. S. Coast Guard Headquarters
Office of Research and Development
Washington, D. C. 20590

13. Type of Report and Period Covered
Final Report.
July 77--June 78

14. Sponsoring Agency Code
G-DSA-1

15. Supplementary Notes The U. S. Coast Guard's Research and Development technical representatives for the work described herein were Larry J. Olson and David A. Walden.

16. Abstract

A mathematical model has been formulated to simulate the motion of a river barge flotilla maneuvering in an inland waterway. The mathematical model includes the inertial and hydrodynamic forces due to the motion of the flotilla through the water, as well as applied forces due to rudder and propeller action, wind and current effects. The layout of virtually any river channel boundaries, along with arbitrary wind and current speed and direction (including cross currents, non-uniform currents and eddys) can be easily described in the model coordinate system. In the present formulation, bank suction and shallow water effects are neglected.

The mathematical model has been implemented in a FORTRAN computer program. The program is written to run in a "conversational" mode, periodically displaying the present status of the simulation to the user, and prompting for input of updated rudder and speed commands. The program has been structured to permit differnt tow characteristics and steering strategies to be evaluated with a minimum of difficulty. A sample program input deck and output listing are shown.

This volume includes a program listing.

17. Key Words	18. Distribution Statement
Towboat Maneuvering Computer Simulation	Document is available to the public through the National Technical Information Service, Springfield, Va. 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified		

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

METRIC CONVERSION FACTORS

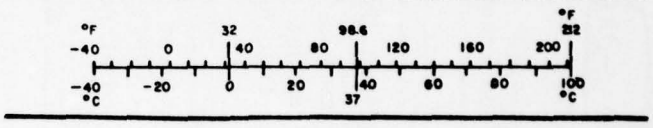
Approximate Conversions to Metric Measures

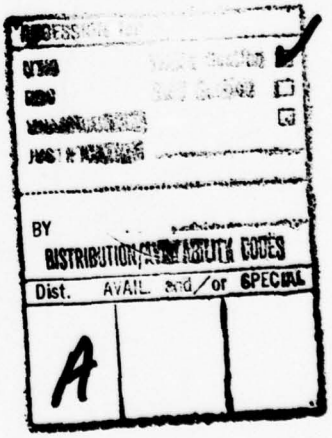
Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
	AREA			
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
	MASS (weight)			
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
	VOLUME			
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
	TEMPERATURE (exact)			
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
	AREA			
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
	MASS (weight)			
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
	VOLUME			
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
	TEMPERATURE (exact)			
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F





I. INTRODUCTION

A mathematical model suitable for simulation of bridge collision incidents involving inland waterway tows has been developed, as described in the accompanying report[1]. The mathematical model has been implemented in a FORTRAN computer program, and is designed to run in a time sharing or dedicated processer environment, with interaction from the user throughout the simulation. The input data file format is simple and consistent, and all interaction with the program is in response to on-line prompts. The structure of the input data file is described in the next section, and a discussion of how to run the program and interpret the output is given in the following section. The discussion is presented in relation to a sample case, and the corresponding input and output formats are illustrated in the Appendices. A program listing is also appended to this report.

II. INPUT DATA FILE DESCRIPTION

The input data file format has been structured to be as convenient as possible from the users prospective. Each line of the data file is coded, to identify the type of data contained on that line, and all data lines are arranged in the same format. Therefore, the user can check very easily whether or not all of the required data elements have been included in the data file.

The program is structured to enable several simulation cases to be executed consecutively from the same data file. To execute several cases consecutively it is necessary only to specify the data values for each case which differ from the preceeding case. Furthermore, the requirements for ordering the lines in the data file are minimal; the first line of the file must contain the text of a heading that appears on the output, the last line of data for each case must be a case delimiter, and the last line of the data file must be a run delimiter. The lines of data within each case may be arranged in any order. Cases are executed sequentially, in the order in which they appear in the data file.

---

[1] Petrie, G.L., "Simulation of Bridge Collision Incidents", HMC Report 7764, to U.S. Coast Guard under Contract No. DOT-CG-72402-A, May 1978

The program assumes the first line in the data file has the format (20A4). All remaining lines are assumed to have the format (A4, 1X, I1, I3, IX, 7F10.5). The characters and values in each field are read, one line at a time, and assigned to the following temporary variables:

Column	Type	Name
1- 4	Alpha	LABEL
6	Integer	ITAG
7- 9	Integer	JTAG
11-20	Fixed	TEMP(1)
21-30	Fixed	TEMP(2)
31-40	Fixed	TEMP(3)
41-50	Fixed	TEMP(4)
51-60	Fixed	TEMP(5)
61-70	Fixed	TEMP(6)
71-80	Fixed	TEMP(7)

The value of LABEL in each line specifies the type of data contained in that line. The permissible character strings for LABEL, and the type of data to which each corresponds, are:

BARG - data describes the characteristics of the barge array or the tugboat,

INIT - data describes the initial conditions (velocity and orientation) of the flotilla,

ROUT - data describes a segment of a route,

COEF - data specifies values for hydrodynamic coefficients,

STER - data specifies steering criteria parameters,

TRIP - case delimiter; causes execution of a case to begin,

QUIT - run delimeter; causes program execution to terminate.

A typical data file may consist of a heading (TITLE) line, five BARG lines, two INIT lines, a STER line, four COEF lines and a number of ROUT lines. The input for the first case is terminated with a TRIP line. Subsequent cases need contain only the lines that are changed from the preceeding case, and are each (including the last) terminated by a TRIP line. The last line in the data file is a QUIT line. A typical input data file is shown in Appendix A. The content of each type of data line is discussed below.

Heading

The first line of the data file; its content is printed as a heading on the output.

Barge and Tugboat Characteristics

Five lines of data are required to describe the dimensions and physical characteristics of the flotilla, its rudder and its propeller.

Name	Units	Entry
LABEL	-----	BARG
ITAG	-----	1
JTAG	-----	NBARW, number of barges wide
TEMP(1)	ft	TOWWID, width of barge array
TEMP(2)	ft	BOTWID, width of tugboat

Name	Units	Entry
LABEL	-----	BARG
ITAG	-----	2
JTAG	-----	NBARL, number of barges long
TEMP(1)	ft	TOWLEN, length of barge array
TEMP(2)	ft	BOTLEN, length of tugboat
TEMP(3)	ft	CGTOW, LCG of barge array
TEMP(4)	ft	CGBOT, LCG of tugboat
TEMP(5)	ft	TOWK, gyradius of barge array
TEMP(6)	ft	BOTK, gyradius of tugboat

Name	Units	Entry
LABEL	-----	BARG
ITAG	-----	3
JTAG	-----	0
TEMP(1)	ft	TOWDRF, draft of barge array
TEMP(2)	ft	BOTDRF, draft of tugboat
TEMP(3)	-----	TOWBC, block coef. of barge array
TEMP(4)	-----	BOTBC, block coef. of tugboat

Name	Units	Entry
LABEL	-----	BARG
ITAG	-----	4
JTAG	-----	NPROP, number of propellers. If NPROP > 0 four quadrant operation is assumed. If NPROP < 0, NPROP = |NPROP| and only first quadrant operation is possible
TEMP(1)	hp	SHP, horsepower per shaft
TEMP(2)	rpm	RPMAX, maximum propeller speed
TEMP(3)	ft	OFSET, shaft offset from centerline
TEMP(4)	ft^2	ARUD, rudder area
TEMP(5)	rad	DELMAX, maximum rudder angle
TEMP(6)	rad/sec	DLDTMX, maximum rudder rate

Name	Units	Entry
LABEL	-----	BARG
ITAG	-----	5
JTAG	-----	NPROP
TEMP(1)	ft	DPROP, propeller diameter
TEMP(2)	ft	PITCH, propeller pitch
TEMP(3)	-----	ARAT, propeller blade area ratio
TEMP(4)	-----	WFRAC, wake fraction
TEMP(5)	-----	TDDUC, thrust deduction factor

Initial Conditions

One line is required to specify the initial velocity and orientation of the flotilla as it enters a curved segment. Another line specifies the integration step size and error control parameters.

Name	Units	Entry
LABEL	-----	INIT
ITAG	-----	0
JTAG	-----	INSEG, initial segment number
TEMP(2)	ft/sec	SPDIN, initial speed of flotilla
TEMP(4)	rad	GAMIN, initial velocity vector angle
TEMP(5)	rad	HEADIN, initial yaw angle
TEMP(6)	rad/sec	CDOTIN, initial yaw rate
TEMP(7)	-----	DRADIN, initial radial offset factor

Name	Units	Entry
LABEL	-----	INIT
ITAG	-----	1
JTAG	-----	NCUTS, integration step size limiter
TEMP(1)	sec	STEP, integration interval
TEMP(2)	sec	FIRSTP, nominal integration step size
TEMP(3)	-----	EPS, relative error limit
TEMP(4)	-----	AB, absolute error limit

Route Description

One line contains the number of segments, NSEG, used to describe a route. An additional number, NSEG, of lines contain the characteristics of each segment.

Name	Units	Entry
LABEL	-----	ROUT
ITAG	-----	0
JTAG	-----	NSEG, number of last route segment

Name	Units	Entry
LABEL	-----	ROUT
ITAG	-----	ISUB, sub-segment number
JTAG	-----	ISEG, segment number
TEMP(1)	deg	SANG (ISUB, ISEG), angular extent of subsegment (ISUB - 1), if ITAG > 1
	mph	SWIND (ISEG), wind speed, if ITAG = 1
TEMP(2)	ft	SDAT (ISUB, ISEG, 1), radial distance to inside wall of channel
TEMP(3)	ft	SDAT (ISUB, ISEG, 2), radial distance to outside wall of channel
TEMP(4)	ft/sec	CUR (ISUB, ISEG, 1), current speed at inside wall of channel
TEMP(5)	ft/sec	CUR (ISUB, ISEG, 2), current speed at outside wall of channel
TEMP(6)	ft/sec	CUR (ISUB, ISEG, 3), cross current speed
TEMP(7)	deg	DWIND (ISEG), wind direction if ITAG = 1

Hydrodynamic Coefficients

As many as four lines are required to specify values for the hydrodynamic coefficients. If a bowthruster is included in the flotilla, three additional lines are required to satisfy its characteristics.

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	1
JTAG	-----	1
TEMP(1)	-----	AA(2), coefficient a_2
TEMP(2)	-----	AA(3), coefficient a_3
TEMP(3)	-----	AA(9), coefficient a_9
TEMP(4)	-----	AA(10), coefficient a_{10}

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	1
JTAG	-----	2
TEMP(1)	-----	AA(1), coefficient a_1
TEMP(2)	-----	AA(5), coefficient a_5
TEMP(3)	-----	AA(6), coefficient a_6
TEMP(4)	-----	AA(7), coefficient a_7

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	2
JTAG	-----	1
TEMP(1)	-----	BB(2), coefficient b_2
TEMP(2)	-----	BB(3), coefficient b_3
TEMP(3)	-----	BB(9), coefficient b_9
TEMP(4)	-----	BB(10), coefficient b_{10}

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	2
JTAG	-----	2
TEMP(1)	-----	BB(1), coefficient b_1
TEMP(2)	-----	BB(5), coefficient b_5
TEMP(3)	-----	BB(6), coefficient b_6
TEMP(4)	-----	BB(7), coefficient b_7

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	4
JTAG	-----	NBTSPD; the number of data points describing the bowthruster performance curve; if NBTSPD = 0, no bowthruster is included; required only if one was included in a previous case.
TEMP(1)	lb	BTMAX, maximum bowthruster force
TEMP(2)	ft	BTPOS, position of bowthruster
TEMP(3)	-----	BTGAIN, lateral thrust command parameter

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	5
JTAG	-----	1
TEMP(1)	ft/sec	BTSPD(1), speed for data point 1
TEMP(2)	ft/sec	BTSPD(2), speed for data point 2
TEMP(3)	ft/sec	BTSPD(3), speed for data point 3
TEMP(4)	ft/sec	BTSPD(4), speed for data point 4
TEMP(5)	ft/sec	BTSPD(5), speed for data point 5
TEMP(6)	ft/sec	BTSPD(6), speed for data point 6
TEMP(7)	ft/sec	BTSPD(7), speed for data point 7

Name	Units	Entry
LABEL	-----	COEF
ITAG	-----	5
JTAG	-----	2
TEMP(1)	lb	BTHRUS(1), thrust at BTSPD (1)
TEMP(2)	lb	BTHRUS(2), thrust at BTSPD (2)
TEMP(3)	lb	BTHRUS(3), thrust at BTSPD (3)
TEMP(4)	lb	BTHRUS(4), thrust at BTSPD (4)
TEMP(5)	lb	BTHRUS(5), thrust at BTSPD (5)
TEMP(6)	lb	BTHRUS(6), thrust at BTSPD (6)
TEMP(7)	lb	BTHRUS(7), thrust at BTSPD (7)

Steering Criteria

One line is required to specify values for the rudder control parameters. This data is not required for solution by interpolation.

Name	Units	Entry
LABEL	-----	STER
ITAG	-----	0
JTAG	-----	0
TEMP(1)	-----	BOWCLR, initial bow clearance factor

Case Delimiter

One line is required to initiate execution of a case and to specify the solution option selected. This line must immediately follow the last line of data for each case.

Name	Units	Entry
LABEL	-----	TRIP

Run Delimiter

The last line of the data file; it causes the program execution to terminate.

Name	Units	Entry
LABEL	-----	QUIT

III. PROGRAM EXECUTION AND OUTPUT DESCRIPTION

Prior to execution of the computer simulation, it is necessary to set up an input data file, which will specify the physical and hydrodynamic characteristics of the flotilla, its initial position, orientation and velocity, and the physical characteristics of the channel. The format of the input data file, and the ordering of the variables within each line, is described in the preceeding section. The data requirements in most cases are entirely self explanatory, however, certain conventions and options which require supplementary explanation are described in greater detail below.

In the BARG 5 data line, the second integer entry, JTAG, specified the number of blades per propeller. However, if this entry is made negative, the absolute value will be assumed, but it will serve as a flag to direct the program to use an alternate data file for the propeller characteristics.

The normal data file has four quadrant data for a specific propeller configuration (blade area ratio, pitch/diameter ratio, etc.). The alternate data file has a more generalized structure, where blade area and pitch/diameter can assume any values, however, the data is only applicable to ahead operation with forward propeller rotation. The option is desirable, however, if the effect of variations in propeller characteristics are to be evaluated.

The physical characteristics of the channel are described by the set of ROUT data, as described previously. The sign convention assumed for the current direction is that the current runs opposed to the barge direction; that is, for a segment where $\alpha_o > 0$, the current runs clockwise around the origin of the segment. When $\alpha_o < 0$, the current runs conterclockwise around the origin of the segment. The cross current is assumed to run radially outward away from the origin of the segment. To reverse any of these directions, simply specify the appropriate velocity component to be negative. Note particularly that the current along the inside and outside wall of the channel may have opposed signs, thereby representing a current reversal or eddy.

If a bowthruster is to be integrated into the flotilla, its location and maximum (zero-speed) thrust are given in the COEF 4 line of data. The thrust versus speed characteristics are given in two COEF 5 lines, in terms of points describing a curve such as shown in Figure 1. Note that the ordinates of the curve are nomalized by the zero speed thrust, thus different bowthruster ratings can be evaluated by changing only a single line of data.

The integration control parameters, given in the INIT 1 data set specify the time intervals of the interpolation. The step size determines how frequently the solution summary is printed out and the rudder commands are requested. It is most convenient to keep the step size as a power of two, since it may be halved successively by the program to meet the error control limits. The values used in the Sample Input file, Appendix A, are good nominal values for most usage, although others can be substituted if desired.

The INIT 0 and STER data sets specify the initial conditions of the flotilla. All values are self evident except for DRADIN and BOWCLR. These two parameters define the initial radial offset, d_R, of the barge from the center line of the channel by the expression

$$d_R = \text{DRADIN} * (\text{BOWCLR} * W_S - (W_S - W_T) / 2)$$

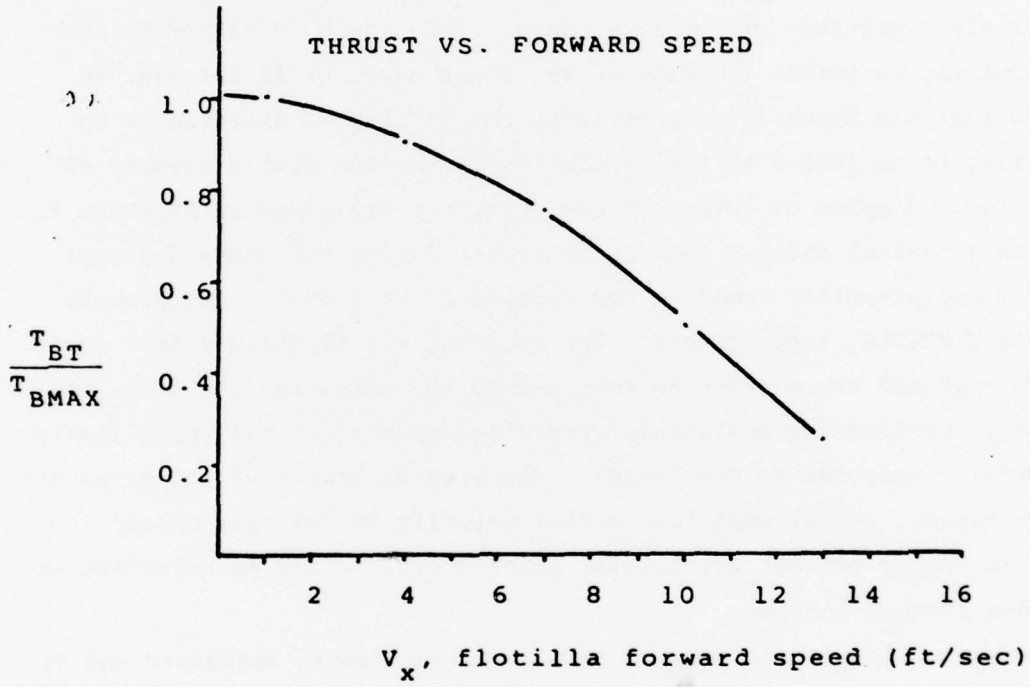

THRUST VS. FORWARD SPEED
1.0
0.8
0.6
0.4
0.2
$\frac{T_{BT}}{T_{BMAX}}$
2
4
6
8
10
12
14
16
V_x, flotilla forward speed (ft/sec)

Figure 1.

where

W_S = the segment width at its beginning

W_T = the flotilla width

The values given in the sample input data file will serve to place the flotilla approximately in the middle of the channel initially.

Once the input data file is set up, the simulation program can be executed. The program will read all data for the first case, down to the TRIP delimiter line. Execution of the case then begins, and the first output is simply a printout of the input data. This can be directed to the users terminal, to permit checking of the input data, or if the user is confident that his input file is correct, the listing is directed to an output file, to be listed at the conclusion of the run with a summary of the maneuver. Samples of both of these printouts are given in Appendix B, along with a typical display that is generated during the command prompt.

Rudder and propeller commands are entered in response to the prompts RUDDER and THROTTLE, respectively. The steering and flanking rudder positions (in degrees) are entered in response to the first and the port, starboard and centerline (as applicable),propeller speed (r.p.m.),upper limits are entered in response to the latter. The present status of the barge motion (its r.p.m., rudder position, radial velocity in feet per second and yaw rate in degree per second) is also printed out, to aid in selecting an appropriate command setting.

The r.p.m. is printed out for each propeller, port, starboard and centerline respectively. If the towboat is twin screw, the centerline propeller r.p.m. will be shown as zero. In the case of a single screw towboat, the r.p.m. will be displayed by the label 'RPM-P'. The radial velocity, labelled 'V-RAD', represents the velocity of the flotilla outward from the center of the curve. The relative yaw rate, labelled 'V-YAW', represents the rate at which the flotilla is turning in relation to the radially directed vector. The rudder positions, 'S-RUD' and 'F-RUD' indicate the present position, in degrees, or the steering and flanking rudders, respectively.

Two options are available through the rudder command. If the steering rudder position is greater than 90^{o}, the simulation will "back-track" to the preceeding time step, so that the most recent command can be altered. If the rudder position is less than -90^{o}, the simulation will "move" the flotilla back to the beginning of the present segment, and reinitialize

its motion to the values that were applicable at that time. Thus, if a maneuver is represented by several major segments, each stage can be repetitively analyzed before proceeding with the simulation, without re-running the entire maneuver from the start. This is a most convenient feature for analyzing a maneuver that requires, say, a sharp turn followed by a narrow passage and another bend. Each successvie stage can be exhaustively simulated to determine the best strategy for approaching the following segment.

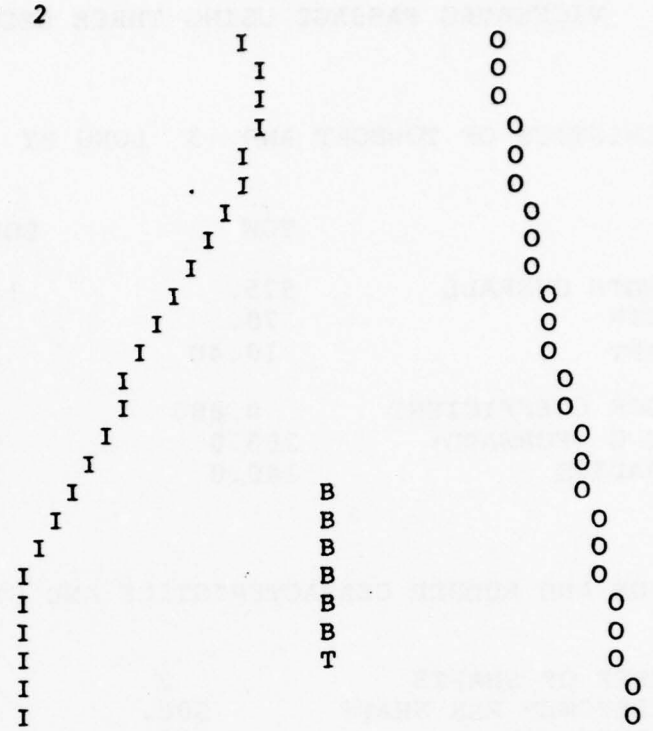
The simulation thus offers a wide range of flexibility in selecting different barge configurations and channel descriptions. The input data requirements, both from the data file and from the interactive command mode, are simple and convenient. The periodic display of the barge position, and interrogation for rudder and throttle commands enables the user to develop a "feel" for how the barge responds to commands and environmental effects. Thus, the simulation offers a beneficial means of evaluating causes and counter measures relevant to inland waterway bridge collisions.

APPENDIX A

```
VICKSBURG PASSAGE USING THREE SEGMENTS.  SAMPLE CASE.
BARG 1  2       70.0       70.0
BARG 2  3      574.5      112.5      283.0       56.0      140.0       28.0
BARG 3  0       10.4       13.8       0.88       0.80
BARG 4  2      500.0      250.0       10.0       50.0       50.0        0.6       0.0
BARG 5 -4        9.5        7.5       0.85        0.0        0.0
INIT 0  2      100.0        9.0        0.0        0.0        0.0        0.0       1.
INIT 1 20        8.0        1.0       0.01       0.01
STER 0  0       0.50       0.06       0.06      0.001       0.05        4.0
ROUT 0  3
ROUT 1  1        0.0       5200.      7800.
ROUT 2  1       20.0       5000.      6800.
ROUT 1  2        0.0       5000.      6800.
ROUT 2  2       15.0       5900.      6700.
ROUT 3  2       10.0       6200.      7000.
ROUT 4  2       20.0       6000.      8200.
ROUT 5  2       20.0       8200.      9400.
ROUT 6  2       15.0       8200.      9300.
ROUT 7  2       15.0       8000.      9300.
ROUT 8  2       15.0       7400.      9000.
ROUT 1  3        0.0       7400.      9000.
ROUT 2  3       30.0       6000.      9000.
COEF 1  1   -0.00058    -0.0011        0.0        0.0
COEF 2  1    -0.0033    -0.0011        0.0        0.0
COEF 1  2        0.0    -0.0072    0.00026    -0.0070
COEF 2  2        0.0     -0.012    -0.0081     -0.016
TRIP 2
QUIT
```

APPENDIX B

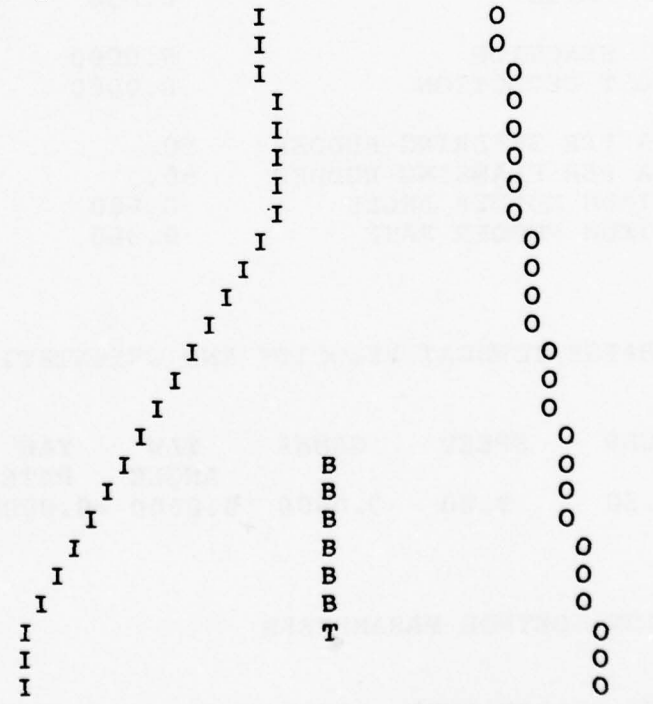
0.0 PERCENT THRU SEGMENT 2



SPEED 6.1
RPM-P 0.0
RPM-S 0.0
-CL 0.0
V-RAD 0.0
V-YAW 0.1
S-RUD 0.0
F-RUD 0.0

RUDDER=
5. 5.
THROTTLE=
150. 150.

3.2 PERCENT THRU SEGMENT 2



SPEED 6.1
RPM-P130.0
RPM-S130.0
-CL 0.0
V-RAD 0.0
V-YAW 0.1
S-RUD 5.0
F-RUD 5.0

RUDDER=
5. 5.
THROTTLE=
150. 150.

VICKSBURG PASSAGE USING THREE SEGMENTS. SAMPLE CASE.

CHARACTERISTICS OF TOWBOAT AND 3 LONG BY 2 WIDE BARGE TOW

	TOW	BOAT
LENGTH OVERALL	575.	113.
WIDTH	70.	70.
DRAFT	10.40	13.80
BLOCK COEFFICIENT	0.880	0.800
L C G (FORWARD)	283.0	56.0
GYRADIUS	140.0	28.0

PROPULSION AND RUDDER CHARACTERISTICS AND STEERING CRITERIA

NUMBER OF SHAFTS	2
HORSEPOWER PER SHAFT	500.
MAXIMUM RPM	250.
SHAFT OFFSET FROM CL	10.
BLADES PER PROPELLER	-4
DIAMETER	9.5
PITCH	7.5
AREA RATIO	0.850
WAKE FRACTION	0.0000
THRUST DEDUCTION	0.0000
AREA PER STEERING RUDDER	50.
AREA PER FLANKING RUDDER	50.
MAXIMUM RUDDER ANGLE	0.600
MAXIMUM RUDDER RATE	0.060

INITIAL BARGE/TOWBOAT VELOCITY AND ORIENTATION

BOWCLR	SPEED	GAMMA	YAW ANGLE	YAW RATE	RADIAL OFFSET
0.50	9.00	0.0000	0.0000	0.0000	1.000

INTEGRATION CONTROL PARAMETERS

NCUTS	FIRST-STEP	STEP-SIZE	REL-ERROR	ABS-ERROR
20	1.0000	8.0000	0.01000000	0.01000000

INTEGRATION THROUGH SEGMENT NUMBER 2

STEP	TIME	V(ANG)	ALPHA	V(HEAD)	PSI	V(RAD)	RADIUS	BETA	GAMA	DELTA	RPM	CLIN	XLIN	CLOUT	XLOUT
1	8.	0.154E-02	0.012	0.873E-03	0.0093	0.2013	5936.	-0.0127	0.0221	0.0873	130.	239.	1570.	523.	1865.
2	16.	0.155E-02	0.025	0.479E-03	0.0146	0.4037	5938.	-0.0291	0.0437	0.0873	130.	239.	1570.	523.	1865.
3	24.	0.157E-02	0.037	0.219E-03	0.0174	0.5976	5942.	-0.0467	0.0640	0.0873	130.	239.	1570.	523.	1865.
4	32.	0.158E-02	0.050	0.336E-04	0.0183	0.7780	5948.	-0.0641	0.0824	0.0873	130.	239.	1570.	523.	1865.
5	40.	0.160E-02	0.062	-0.111E-03	0.0180	0.9422	5955.	-0.0809	0.0989	0.0873	130.	239.	1570.	523.	1865.
6	48.	0.161E-02	0.075	-0.232E-03	0.0166	1.0891	5963.	-0.0966	0.1132	0.0873	130.	193.	1178.	557.	1865.
7	56.	0.162E-02	0.088	-0.339E-03	0.0143	1.2181	5972.	-0.1112	0.1255	0.0873	130.	193.	1178.	557.	1865.
8	64.	0.162E-02	0.101	-0.439E-03	0.0112	1.3292	5982.	-0.1247	0.1359	0.0873	130.	193.	1178.	557.	1865.
9	72.	0.163E-02	0.114	-0.532E-03	0.0073	1.4227	5993.	-0.1371	0.1444	0.0873	130.	193.	1178.	557.	1865.
10	80.	0.164E-02	0.127	-0.621E-03	0.0027	1.4989	6005.	-0.1485	0.1512	0.0873	130.	193.	1178.	557.	1865.
11	88.	0.165E-02	0.140	0.577E-03	0.0030	1.3795	6016.	-0.1351	0.1381	-0.1745	135.	105.	1865.	633.	883.
12	96.	0.166E-02	0.154	0.126E-02	0.0106	1.2592	6027.	-0.1145	0.1251	-0.1745	135.	105.	1865.	633.	883.
13	104.	0.167E-02	0.167	0.171E-02	0.0226	1.1567	6037.	-0.0915	0.1141	-0.1745	135.	105.	1865.	633.	883.
14	112.	0.168E-02	0.180	0.204E-02	0.0376	1.0794	6046.	-0.0681	0.1057	-0.1745	135.	105.	1865.	633.	883.
15	120.	0.169E-02	0.194	0.232E-02	0.0551	1.0299	6054.	-0.0450	0.1001	-0.1745	135.	105.	1865.	633.	883.
16	128.	0.170E-02	0.207	0.255E-02	0.0746	1.0089	6062.	-0.0228	0.0974	-0.1745	135.	118.	1472.	605.	491.
17	136.	0.171E-02	0.221	0.276E-02	0.0959	1.0162	6070.	-0.0015	0.0974	-0.1745	135.	118.	1472.	605.	491.
18	144.	0.172E-02	0.235	0.295E-02	0.1188	1.0514	6078.	0.0186	0.1002	-0.1745	135.	118.	1472.	605.	491.
19	152.	0.173E-02	0.249	0.312E-02	0.1431	1.1139	6087.	0.0376	0.1055	-0.1745	135.	118.	1472.	605.	491.
20	160.	0.173E-02	0.262	0.327E-02	0.1686	1.2029	6096.	0.0554	0.1133	-0.1745	135.	118.	1472.	605.	491.
21	168.	0.174E-02	0.276	0.275E-02	0.1925	1.4098	6107.	0.0602	0.1322	0.0000	135.	172.	981.	532.	1865.
22	176.	0.173E-02	0.290	0.249E-02	0.2133	1.6338	6119.	0.0606	0.1527	0.0000	135.	172.	981.	532.	1865.
23	184.	0.173E-02	0.304	0.236E-02	0.2327	1.8634	6133.	0.0591	0.1736	0.0000	135.	172.	981.	532.	1865.
24	192.	0.173E-02	0.318	0.229E-02	0.2513	2.0933	6149.	0.0567	0.1945	0.0000	135.	172.	981.	532.	1865.
25	200.	0.172E-02	0.332	0.223E-02	0.2693	2.3211	6166.	0.0541	0.2152	0.0000	135.	172.	981.	532.	1865.
26	208.	0.171E-02	0.345	0.219E-02	0.2870	2.5455	6186.	0.0515	0.2356	0.0000	135.	162.	-393.	451.	1865.
27	216.	0.171E-02	0.359	0.216E-02	0.3044	2.7660	6207.	0.0489	0.2555	0.0000	135.	162.	-393.	451.	1865.
28	224.	0.170E-02	0.373	0.212E-02	0.3216	2.9822	6230.	0.0464	0.2751	0.0000	135.	162.	-393.	451.	1865.
29	232.	0.168E-02	0.386	0.209E-02	0.3384	3.1941	6255.	0.0441	0.2943	0.0000	135.	162.	-393.	451.	1865.
30	240.	0.167E-02	0.400	0.206E-02	0.3550	3.4014	6281.	0.0418	0.3132	0.0000	135.	162.	-393.	451.	1865.
31	248.	0.165E-02	0.413	0.136E-02	0.3684	3.6918	6310.	0.0282	0.3402	0.1745	135.	140.	-491.	379.	1865.
32	256.	0.163E-02	0.426	0.933E-03	0.3775	3.9713	6340.	0.0111	0.3663	0.1745	135.	140.	-491.	379.	1865.
33	264.	0.161E-02	0.439	0.637E-03	0.3837	4.2303	6373.	-0.0068	0.3905	0.1745	135.	140.	-491.	379.	1865.
34	272.	0.159E-02	0.452	0.406E-03	0.3878	4.4655	6408.	-0.0246	0.4124	0.1745	135.	140.	-491.	379.	1865.
35	280.	0.157E-02	0.465	0.211E-03	0.3903	4.6758	6444.	-0.0418	0.4320	0.1745	135.	140.	-491.	379.	1865.
36	288.	0.156E-02	0.477	0.368E-03	0.3927	4.8184	6482.	-0.0524	0.4451	0.0873	135.	162.	-294.	373.	1865.
37	296.	0.154E-02	0.490	0.397E-03	0.3958	4.9406	6522.	-0.0605	0.4563	0.0873	135.	162.	-294.	373.	1865.
38	304.	0.153E-02	0.502	0.372E-03	0.3989	5.0482	6561.	-0.0672	0.4661	0.0873	135.	162.	-294.	373.	1865.
39	312.	0.152E-02	0.514	0.324E-03	0.4017	5.1440	6602.	-0.0732	0.4749	0.0873	135.	162.	-294.	373.	1865.
40	320.	0.150E-02	0.526	0.269E-03	0.4041	5.2296	6644.	-0.0786	0.4827	0.0873	135.	162.	-294.	373.	1865.
41	328.	0.149E-02	0.538	0.211E-03	0.4060	5.3059	6686.	-0.0837	0.4897	0.0873	135.	347.	-491.	391.	1865.
42	336.	0.148E-02	0.550	0.154E-03	0.4075	5.3735	6729.	-0.0884	0.4959	0.0873	135.	347.	-491.	391.	1865.
43	344.	0.146E-02	0.562	0.992E-04	0.4085	5.4330	6772.	-0.0929	0.5013	0.0873	135.	347.	-491.	391.	1865.
44	352.	0.145E-02	0.573	0.464E-04	0.4091	5.4848	6816.	-0.0970	0.5061	0.0873	135.	347.	-491.	391.	1865.
45	360.	0.144E-02	0.585	-0.393E-05	0.4092	5.5293	6860.	-0.1009	0.5102	0.0873	135.	347.	-491.	391.	1865.
46	368.	0.143E-02	0.596	-0.377E-03	0.4076	5.6093	6904.	-0.1102	0.5177	0.1745	135.	676.	-491.	431.	1865.
47	376.	0.141E-02	0.608	-0.604E-03	0.4036	5.6783	6949.	-0.1207	0.5242	0.1745	135.	676.	-491.	431.	1865.
48	384.	0.140E-02	0.619	-0.765E-03	0.3981	5.7316	6995.	-0.1312	0.5293	0.1745	135.	676.	-491.	431.	1865.
49	392.	0.139E-02	0.630	-0.895E-03	0.3914	5.7682	7041.	-0.1413	0.5327	0.1745	135.	676.	-491.	431.	1865.
50	400.	0.138E-02	0.641	-0.101E-02	0.3838	5.7880	7087.	-0.1508	0.5346	0.1745	135.	676.	-491.	431.	1865.

STEP	TIME	V(ANG)	ALPHA	V(HEAD)	PSI	V(RAD)	RADIUS	BETA	GAMA	DELTA	RPM	CLIN	XLIN	CLOUT	XLOUT
51	408.	0.135E-02	0.652	-0.508E-03	0.3780	5.6244	7133.	-0.1517	0.5297	0.0000	80.	977.	-491.	378.	1865.
52	416.	0.132E-02	0.663	-0.285E-03	0.3749	5.4606	7177.	-0.1491	0.5240	0.0000	80.	977.	-491.	378.	1865.
53	424.	0.129E-02	0.673	-0.169E-03	0.3732	5.3053	7220.	-0.1453	0.5185	0.0000	80.	977.	-491.	378.	1865.
54	432.	0.126E-02	0.683	-0.963E-04	0.3721	5.1611	7262.	-0.1412	0.5133	0.0000	80.	977.	-491.	378.	1865.
55	440.	0.123E-02	0.693	-0.427E-04	0.3716	5.0280	7303.	-0.1370	0.5086	0.0000	80.	977.	-491.	378.	1865.
56	448.	0.121E-02	0.703	0.113E-05	0.3714	4.9056	7343.	-0.1329	0.5044	0.0000	80.	1228.	-491.	296.	1865.
57	456.	0.119E-02	0.713	0.392E-04	0.3716	4.7929	7381.	-0.1290	0.5006	0.0000	80.	1228.	-491.	296.	1865.
58	464.	0.116E-02	0.722	0.732E-04	0.3720	4.6893	7419.	-0.1251	0.4972	0.0000	80.	1228.	-491.	296.	1865.
59	472.	0.114E-02	0.731	0.104E-03	0.3727	4.5938	7456.	-0.1214	0.4942	0.0000	80.	1228.	-491.	296.	1865.
60	480.	0.112E-02	0.740	0.132E-03	0.3737	4.5059	7493.	-0.1179	0.4916	0.0000	80.	1228.	-491.	296.	1865.
61	488.	0.110E-02	0.749	-0.321E-03	0.3728	4.4858	7529.	-0.1235	0.4963	0.3490	80.	1216.	1865.	227.	1865.
62	496.	0.108E-02	0.758	-0.588E-03	0.3690	4.4680	7565.	-0.1320	0.5010	0.3490	80.	1216.	1865.	227.	1865.
63	504.	0.106E-02	0.767	-0.766E-03	0.3636	4.4468	7600.	-0.1416	0.5052	0.3490	80.	1216.	1865.	227.	1865.
64	512.	0.104E-02	0.775	-0.899E-03	0.3569	4.4198	7636.	-0.1515	0.5084	0.3490	80.	1216.	1865.	227.	1865.
65	520.	0.102E-02	0.783	-0.101E-02	0.3493	4.3862	7671.	-0.1614	0.5107	0.3490	80.	1216.	1865.	227.	1865.
66	528.	0.100E-02	0.791	-0.110E-02	0.3408	4.3460	7706.	-0.1711	0.5119	0.3490	80.	1096.	1865.	227.	1865.
67	536.	0.988E-03	0.799	-0.119E-02	0.3316	4.2992	7741.	-0.1805	0.5121	0.3490	80.	1096.	1865.	227.	1865.
68	544.	0.973E-03	0.807	-0.127E-02	0.3218	4.2464	7775.	-0.1895	0.5113	0.3490	80.	1096.	1865.	227.	1865.
69	552.	0.960E-03	0.815	-0.135E-02	0.3113	4.1877	7808.	-0.1983	0.5095	0.3490	80.	1096.	1865.	227.	1865.
70	560.	0.947E-03	0.822	-0.142E-02	0.3002	4.1236	7842.	-0.2066	0.5068	0.3490	80.	1096.	1865.	227.	1865.
71	568.	0.937E-03	0.830	-0.126E-02	0.2895	4.0247	7874.	-0.2100	0.4995	0.1745	80.	934.	1865.	286.	1865.
72	576.	0.927E-03	0.837	-0.119E-02	0.2798	3.9243	7906.	-0.2119	0.4917	0.1745	80.	934.	1865.	286.	1865.
73	584.	0.918E-03	0.845	-0.115E-02	0.2705	3.8250	7937.	-0.2132	0.4836	0.1745	80.	934.	1865.	286.	1865.
74	592.	0.909E-03	0.852	-0.114E-02	0.2613	3.7277	7967.	-0.2142	0.4755	0.1745	80.	934.	1865.	286.	1865.
75	600.	0.900E-03	0.859	-0.114E-02	0.2522	3.6328	7997.	-0.2151	0.4673	0.1745	80.	934.	1865.	286.	1865.
76	608.	0.908E-03	0.867	-0.298E-02	0.2350	3.8311	8027.	-0.2491	0.4841	0.6108	125.	760.	1865.	327.	-491.
77	616.	0.916E-03	0.874	-0.394E-02	0.2069	3.9848	8058.	-0.2880	0.4949	0.6108	125.	760.	1865.	327.	-491.
78	624.	0.926E-03	0.881	-0.443E-02	0.1732	4.0704	8090.	-0.3243	0.4976	0.6108	125.	760.	1865.	327.	-491.
79	632.	0.938E-03	0.889	-0.470E-02	0.1366	4.0827	8123.	-0.3555	0.4921	0.6108	125.	760.	1865.	327.	-491.
80	640.	0.949E-03	0.896	-0.489E-02	0.0982	4.0252	8155.	-0.3812	0.4794	0.6108	125.	760.	1865.	327.	-491.
81	648.	0.940E-03	0.904	-0.347E-02	0.0657	3.6698	8186.	-0.3791	0.4448	0.1745	80.	402.	1865.	170.	-491.
82	656.	0.931E-03	0.911	-0.293E-02	0.0404	3.3364	8214.	-0.3710	0.4114	0.1745	80.	402.	1865.	170.	-491.
83	664.	0.922E-03	0.919	-0.265E-02	0.0182	3.0326	8240.	-0.3617	0.3799	0.1745	80.	402.	1865.	170.	-491.
84	672.	0.913E-03	0.926	-0.247E-02	-0.0023	2.7576	8263.	-0.3529	0.3506	0.1745	80.	402.	1865.	170.	-491.
85	680.	0.904E-03	0.933	-0.234E-02	-0.0215	2.5080	8284.	-0.3448	0.3233	0.1745	80.	402.	1865.	170.	-491.
86	688.	0.920E-03	0.941	-0.198E-02	-0.0386	2.2444	8303.	-0.3245	0.2859	0.0000	125.	222.	1668.	73.	-491.
87	696.	0.935E-03	0.948	-0.173E-02	-0.0534	2.0015	8320.	-0.3053	0.2519	0.0000	125.	222.	1668.	73.	-491.
88	704.	0.949E-03	0.956	-0.152E-02	-0.0664	1.7782	8335.	-0.2875	0.2211	0.0000	125.	222.	1668.	73.	-491.
89	712.	0.963E-03	0.963	-0.135E-02	-0.0779	1.5723	8348.	-0.2710	0.1931	0.0000	125.	222.	1668.	73.	-491.
90	720.	0.976E-03	0.971	-0.120E-02	-0.0880	1.3822	8360.	-0.2558	0.1678	0.0000	125.	222.	1668.	73.	-491.
91	728.	0.989E-03	0.979	-0.105E-02	-0.0970	1.2063	8370.	-0.2418	0.1448	0.0000	125.	155.	1374.	68.	-491.
92	736.	0.100E-02	0.987	-0.925E-03	-0.1050	1.0429	8379.	-0.2286	0.1237	0.0000	130.	155.	1374.	68.	-491.
93	744.	0.102E-02	0.995	-0.802E-03	-0.1119	0.8889	8387.	-0.2158	0.1039	0.0000	130.	155.	1374.	68.	-491.
94	752.	0.103E-02	1.003	-0.688E-03	-0.1178	0.7460	8394.	-0.2039	0.0860	0.0000	130.	155.	1374.	68.	-491.
95	760.	0.104E-02	1.011	-0.581E-03	-0.1229	0.6137	8399.	-0.1928	0.0699	0.0000	130.	155.	1374.	68.	-491.
96	768.	0.103E-02	1.020	-0.510E-03	-0.1272	0.5188	8404.	-0.1871	0.0598	0.0000	80.	142.	1080.	145.	-491.
97	776.	0.102E-02	1.028	-0.448E-03	-0.1311	0.4324	8407.	-0.1815	0.0505	0.0000	80.	142.	1080.	145.	-491.
98	784.	0.101E-02	1.036	-0.391E-03	-0.1344	0.3537	8410.	-0.1762	0.0417	0.0000	80.	142.	1080.	145.	-491.
99	792.	0.996E-03	1.044	-0.338E-03	-0.1373	0.2821	8413.	-0.1710	0.0337	0.0000	80.	142.	1080.	145.	-491.
100	800.	0.986E-03	1.052	-0.289E-03	-0.1398	0.2169	8415.	-0.1660	0.0261	0.0000	80.	142.	1080.	145.	-491.

STEP	TIME	V(ANG)	ALPHA	V(HEAD)	PSI	V(RAD)	RADIUS	BETA	GAMA	DELTA	RPM	CLIN	XLIN	CLOUT	XLOUT
101	808.	0.985E-03	1.060	-0.234E-03	-0.1419	0.1474	8416.	-0.1597	0.0178	0.0000	100.	152.	785.	280.	-491.
102	816.	0.984E-03	1.068	-0.181E-03	-0.1436	0.0838	8417.	-0.1537	0.0101	0.0000	100.	152.	785.	280.	-491.
103	824.	0.983E-03	1.076	-0.131E-03	-0.1448	0.0257	8418.	-0.1479	0.0031	0.0000	100.	152.	785.	280.	-491.
104	832.	0.983E-03	1.083	-0.829E-04	-0.1457	-0.0271	8418.	-0.1424	-0.0033	0.0000	100.	152.	785.	280.	-491.
105	840.	0.982E-03	1.091	-0.376E-04	-0.1462	-0.0751	8417.	-0.1371	-0.0091	0.0000	100.	152.	785.	280.	-491.
106	848.	0.999E-03	1.099	0.358E-03	-0.1448	-0.1839	8416.	-0.1229	-0.0219	-0.0873	130.	171.	393.	456.	-491.
107	856.	0.102E-02	1.107	0.632E-03	-0.1408	-0.2859	8414.	-0.1073	-0.0335	-0.0873	130.	171.	393.	456.	-491.
108	864.	0.103E-02	1.115	0.839E-03	-0.1349	-0.3757	8412.	-0.0916	-0.0433	-0.0873	130.	171.	393.	456.	-491.
109	872.	0.105E-02	1.124	0.101E-02	-0.1275	-0.4514	8408.	-0.0762	-0.0512	-0.0873	130.	171.	393.	456.	-491.
110	880.	0.106E-02	1.132	0.115E-02	-0.1188	-0.5122	8405.	-0.0615	-0.0573	-0.0873	130.	171.	393.	456.	-491.
111	888.	0.108E-02	1.141	0.949E-03	-0.1105	-0.5129	8401.	-0.0539	-0.0567	0.0000	130.	202.	98.	685.	-491.
112	896.	0.109E-02	1.149	0.859E-03	-0.1033	-0.4999	8396.	-0.0488	-0.0545	0.0000	130.	202.	98.	685.	-491.
113	904.	0.110E-02	1.158	0.827E-03	-0.0966	-0.4775	8393.	-0.0451	-0.0515	0.0000	130.	202.	98.	685.	-491.
114	912.	0.112E-02	1.167	0.823E-03	-0.0900	-0.4483	8389.	-0.0422	-0.0478	0.0000	130.	202.	98.	685.	-491.
115	920.	0.113E-02	1.176	0.834E-03	-0.0834	-0.4136	8385.	-0.0398	-0.0436	0.0000	130.	202.	98.	685.	-491.
116	928.	0.114E-02	1.185	0.851E-03	-0.0767	-0.3740	8382.	-0.0376	-0.0391	0.0000	130.	220.	-294.	862.	1865.
117	936.	0.115E-02	1.194	0.871E-03	-0.0698	-0.3302	8379.	-0.0356	-0.0342	0.0000	130.	220.	-294.	862.	1865.
118	944.	0.116E-02	1.204	0.892E-03	-0.0627	-0.2823	8377.	-0.0338	-0.0290	0.0000	130.	220.	-294.	862.	1865.
119	952.	0.117E-02	1.213	0.913E-03	-0.0555	-0.2306	8375.	-0.0320	-0.0235	0.0000	130.	220.	-294.	862.	1865.
120	960.	0.118E-02	1.222	0.933E-03	-0.0481	-0.1756	8373.	-0.0304	-0.0177	0.0000	135.	220.	-294.	862.	1865.
121	968.	0.119E-02	1.232	0.956E-03	-0.0406	-0.1179	8372.	-0.0288	-0.0118	0.0000	135.	229.	1374.	785.	1865.
122	976.	0.121E-02	1.241	0.978E-03	-0.0328	-0.0563	8371.	-0.0273	-0.0056	0.0000	135.	229.	1374.	785.	1865.
123	984.	0.122E-02	1.251	0.999E-03	-0.0249	0.0091	8371.	-0.0258	0.0009	0.0000	135.	229.	1374.	785.	1865.
124	992.	0.123E-02	1.261	0.102E-02	-0.0169	0.0781	8372.	-0.0245	0.0076	0.0000	135.	229.	1374.	785.	1865.
125	1000.	0.124E-02	1.271	0.104E-02	-0.0086	0.1504	8372.	-0.0232	0.0145	0.0000	135.	229.	1374.	785.	1865.
126	1008.	0.124E-02	1.281	0.105E-02	-0.0003	0.2260	8374.	-0.0220	0.0217	0.0000	135.	229.	981.	685.	1865.
127	1016.	0.125E-02	1.291	0.107E-02	0.0082	0.3045	8376.	-0.0208	0.0291	0.0000	135.	229.	981.	685.	1865.
128	1024.	0.126E-02	1.301	0.109E-02	0.0169	0.3859	8379.	-0.0197	0.0366	0.0000	135.	229.	981.	685.	1865.
129	1032.	0.126E-02	1.311	0.110E-02	0.0256	0.4700	8382.	-0.0187	0.0443	0.0000	135.	229.	981.	685.	1865.
130	1040.	0.127E-02	1.321	0.111E-02	0.0345	0.5565	8386.	-0.0177	0.0522	0.0000	135.	229.	981.	685.	1865.
131	1048.	0.128E-02	1.331	0.113E-02	0.0434	0.6453	8391.	-0.0168	0.0602	0.0000	135.	231.	589.	584.	1865.
132	1056.	0.128E-02	1.341	0.114E-02	0.0525	0.7363	8397.	-0.0160	0.0684	0.0000	135.	231.	589.	584.	1865.
133	1064.	0.128E-02	1.352	0.115E-02	0.0616	0.8292	8403.	-0.0151	0.0767	0.0000	135.	231.	589.	584.	1865.
134	1072.	0.129E-02	1.362	0.116E-02	0.0708	0.9240	8410.	-0.0144	0.0852	0.0000	135.	231.	589.	584.	1865.
135	1080.	0.129E-02	1.372	0.117E-02	0.0801	1.0205	8418.	-0.0136	0.0937	0.0000	135.	231.	589.	584.	1865.
136	1088.	0.129E-02	1.383	0.117E-02	0.0895	1.1185	8426.	-0.0129	0.1024	0.0000	135.	237.	98.	482.	1865.
137	1096.	0.129E-02	1.393	0.118E-02	0.0989	1.2179	8436.	-0.0123	0.1112	0.0000	135.	237.	98.	482.	1865.
138	1104.	0.129E-02	1.403	0.119E-02	0.1084	1.3186	8446.	-0.0116	0.1200	0.0000	135.	237.	98.	482.	1865.
139	1112.	0.130E-02	1.414	0.119E-02	0.1179	1.4204	8457.	-0.0110	0.1290	0.0000	135.	237.	98.	482.	1865.
140	1120.	0.130E-02	1.424	0.120E-02	0.1275	1.5232	8469.	-0.0105	0.1380	0.0000	135.	237.	98.	482.	1865.
141	1128.	0.129E-02	1.434	-0.120E-03	0.1313	1.8135	8482.	-0.0328	0.1641	0.3490	135.	242.	-294.	381.	1865.
142	1136.	0.129E-02	1.445	-0.879E-03	0.1270	2.0917	8498.	-0.0620	0.1890	0.3490	135.	242.	-294.	381.	1865.
143	1144.	0.128E-02	1.455	-0.137E-02	0.1179	2.3350	8515.	-0.0927	0.2106	0.3490	135.	242.	-294.	381.	1865.
144	1152.	0.128E-02	1.465	-0.171E-02	0.1056	2.5338	8535.	-0.1226	0.2282	0.3490	135.	242.	-294.	381.	1865.
145	1160.	0.127E-02	1.475	-0.199E-02	0.0907	2.6839	8556.	-0.1506	0.2413	0.3490	135.	242.	-294.	381.	1865.
146	1168.	0.127E-02	1.485	-0.129E-02	0.0780	2.6502	8577.	-0.1600	0.2380	0.0873	135.	366.	-491.	359.	1865.
147	1176.	0.127E-02	1.496	-0.102E-02	0.0689	2.5891	8598.	-0.1635	0.2324	0.0873	135.	366.	-491.	359.	1865.
148	1184.	0.127E-02	1.506	-0.914E-03	0.0613	2.5172	8618.	-0.1646	0.2259	0.0873	135.	366.	-491.	359.	1865.
149	1192.	0.127E-02	1.516	-0.873E-03	0.0541	2.4416	8638.	-0.1649	0.2191	0.0873	135.	366.	-491.	359.	1865.
150	1200.	0.127E-02	1.526	-0.857E-03	0.0472	2.3647	8657.	-0.1649	0.2121	0.0873	135.	366.	-491.	359.	1865.

STEP	TIME	V(ANG)	ALPHA	V(HEAD)	PSI	V(RAD)	RADIUS	BETA	GAMA	DELTA	RPM	CLIN	XLIN	CLOUT	XLOUT
151	1208.	0.127E-02	1.536	-0.532E-03	0.0418	2.2417	8676.	-0.1592	0.2011	0.0000	135.	561.	-491.	218.	1865.
152	1216.	0.127E-02	1.546	-0.352E-03	0.0384	2.1244	8693.	-0.1522	0.1905	0.0000	135.	561.	-491.	218.	1865.
153	1224.	0.127E-02	1.557	-0.230E-03	0.0360	2.0175	8710.	-0.1448	0.1809	0.0000	135.	561.	-491.	218.	1865.
154	1232.	0.127E-02	1.567	-0.133E-03	0.0346	1.9219	8726.	-0.1377	0.1723	0.0000	135.	561.	-491.	218.	1865.
155	1240.	0.127E-02	1.577	-0.487E-04	0.0339	1.8377	8741.	-0.1307	0.1646	0.0000	135.	561.	-491.	218.	1865.
156	1248.	0.126E-02	1.587	-0.252E-02	0.0225	2.1306	8757.	-0.1680	0.1904	0.3490	135.	702.	-491.	79.	1865.
157	1256.	0.126E-02	1.597	-0.375E-02	-0.0032	2.3750	8775.	-0.2148	0.2116	0.3490	135.	702.	-491.	79.	1865.
158	1264.	0.126E-02	1.607	-0.442E-02	-0.0361	2.5211	8794.	-0.2602	0.2241	0.3490	135.	702.	-491.	79.	1865.
159	1272.	0.126E-02	1.617	-0.483E-02	-0.0732	2.5552	8815.	-0.3001	0.2269	0.3490	135.	702.	-491.	79.	1865.
160	1280.	0.125E-02	1.627	-0.515E-02	-0.1131	2.4809	8835.	-0.3336	0.2204	0.3490	135.	702.	-491.	79.	1865.
161	1288.	0.125E-02	1.637	-0.379E-02	-0.1478	2.0744	8853.	-0.3335	0.1857	0.1745	135.	811.	294.	206.	1865.
162	1296.	0.124E-02	1.647	-0.336E-02	-0.1762	1.6754	8868.	-0.3273	0.1511	0.1745	135.	811.	294.	206.	1865.
163	1304.	0.123E-02	1.657	-0.317E-02	-0.2022	1.3007	8880.	-0.3203	0.1181	0.1745	135.	811.	294.	206.	1865.
164	1312.	0.123E-02	1.667	-0.305E-02	-0.2270	0.9510	8889.	-0.3139	0.0868	0.1745	135.	811.	294.	206.	1865.
165	1320.	0.122E-02	1.677	-0.295E-02	-0.2510	0.6229	8895.	-0.3082	0.0572	0.1745	135.	811.	294.	206.	1865.
166	1328.	0.122E-02	1.686	-0.229E-02	-0.2717	0.2298	8899.	-0.2929	0.0212	0.0000	135.	909.	98.	253.	-491.
167	1336.	0.121E-02	1.696	-0.194E-02	-0.2885	-0.1280	8899.	-0.2766	-0.0119	0.0000	135.	909.	98.	253.	-491.
168	1344.	0.120E-02	1.706	-0.168E-02	-0.3029	-0.4498	8897.	-0.2610	-0.0419	0.0000	135.	909.	98.	253.	-491.
169	1352.	0.120E-02	1.715	-0.147E-02	-0.3155	-0.7390	8892.	-0.2464	-0.0691	0.0000	135.	909.	98.	253.	-491.
170	1360.	0.120E-02	1.725	-0.128E-02	-0.3264	-0.9999	8885.	-0.2328	-0.0937	0.0000	135.	909.	98.	253.	-491.
171	1368.	0.120E-02	1.735	-0.112E-02	-0.3360	-1.2359	8876.	-0.2201	-0.1159	0.0000	135.	1028.	-196.	205.	-491.
172	1376.	0.119E-02	1.744	-0.969E-03	-0.3444	-1.4499	8865.	-0.2083	-0.1361	0.0000	135.	1028.	-196.	205.	-491.
173	1384.	0.119E-02	1.754	-0.832E-03	-0.3516	-1.6442	8853.	-0.1972	-0.1543	0.0000	135.	1028.	-196.	205.	-491.
174	1392.	0.119E-02	1.763	-0.706E-03	-0.3577	-1.8209	8839.	-0.1868	-0.1709	0.0000	135.	1028.	-196.	205.	-491.
175	1400.	0.119E-02	1.773	-0.590E-03	-0.3629	-1.9814	8824.	-0.1771	-0.1858	0.0000	135.	1028.	-196.	205.	-491.
176	1408.	0.120E-02	1.782	-0.482E-03	-0.3672	-2.1271	8807.	-0.1679	-0.1993	0.0000	135.	1025.	1472.	158.	-491.
177	1416.	0.120E-02	1.792	-0.382E-03	-0.3706	-2.2591	8790.	-0.1592	-0.2114	0.0000	135.	1025.	1472.	158.	-491.
178	1424.	0.120E-02	1.802	-0.288E-03	-0.3733	-2.3785	8771.	-0.1510	-0.2223	0.0000	135.	1025.	1472.	158.	-491.
179	1432.	0.120E-02	1.811	-0.201E-03	-0.3753	-2.4861	8752.	-0.1432	-0.2320	0.0000	135.	1025.	1472.	158.	-491.
180	1440.	0.121E-02	1.821	-0.119E-03	-0.3765	-2.5825	8731.	-0.1359	-0.2406	0.0000	135.	1025.	1472.	158.	-491.
181	1448.	0.121E-02	1.830	-0.419E-04	-0.3772	-2.6685	8710.	-0.1290	-0.2482	0.0000	135.	1004.	1178.	175.	-491.
182	1456.	0.121E-02	1.840	0.307E-04	-0.3772	-2.7446	8689.	-0.1224	-0.2548	0.0000	135.	1004.	1178.	175.	-491.
183	1464.	0.122E-02	1.850	0.993E-04	-0.3767	-2.8113	8667.	-0.1162	-0.2605	0.0000	135.	1004.	1178.	175.	-491.
184	1472.	0.122E-02	1.860	0.164E-03	-0.3756	-2.8691	8644.	-0.1103	-0.2654	0.0000	135.	1004.	1178.	175.	-491.
185	1480.	0.123E-02	1.869	0.226E-03	-0.3741	-2.9185	8621.	-0.1047	-0.2694	0.0000	135.	1004.	1178.	175.	-491.
186	1488.	0.123E-02	1.879	0.284E-03	-0.3720	-2.9598	8597.	-0.0993	-0.2727	0.0000	135.	1011.	785.	241.	-491.
187	1496.	0.124E-02	1.889	0.339E-03	-0.3695	-2.9934	8573.	-0.0943	-0.2753	0.0000	135.	1011.	785.	241.	-491.
188	1504.	0.124E-02	1.899	0.392E-03	-0.3666	-3.0196	8549.	-0.0895	-0.2771	0.0000	135.	1011.	785.	241.	-491.
189	1512.	0.125E-02	1.909	0.442E-03	-0.3633	-3.0387	8525.	-0.0850	-0.2783	0.0000	135.	1011.	785.	241.	-491.
190	1520.	0.125E-02	1.919	0.490E-03	-0.3596	-3.0511	8501.	-0.0807	-0.2789	0.0000	135.	1011.	785.	241.	-491.
191	1528.	0.126E-02	1.929	0.535E-03	-0.3555	-3.0569	8476.	-0.0766	-0.2789	0.0000	135.	1028.	393.	347.	-491.

APPENDIX C

```
10000C***  BARMAN***
10010C***  BARGE MANEUVERING SIMULATION
10020C***  ***IMPLICIT REAL*8 (A-H,O-Z)
10030      COMMON/CMPRNT/IN,LPRT,MONITR
10040      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
10050      CALL BEGIN
10060   10 CONTINUE
10070      CALL INPUT
10080      CALL INITAL
10090      CALL ROUTCK
10100      CALL TRPINT
10110      CALL TRPOUT
10120      GO TO 10
10130      END
10140      SUBROUTINE BEGIN
10150C***  ***IMPLICIT REAL*8 (A-H,O-Z)
10160      COMMON/CMPRNT/IN,LPRT,MONITR
10170      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
10180      COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
10190      COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
10200      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
10210      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
10220      COMMON/CMCNST/RHO,GRAV,PI
10230      COMMON/CMCOEF/A(10),B(10),AA(10),BB(10),COEF(5)
10240      COMMON/CMCLOG/NPOLY,NSTRIP,KORPOL,KORSTR
10250      COMMON/CMBOWT/BTHRUS(7),BTSPD(7),BTMAX,BTPOS,BTGAIN,NBTSPD,IDBT
10260      LOGICAL NPOLY,NSTRIP,KORPOL,KORSTR,IDBT
10270C***   THIS SUBROUTINE ASSIGNS INITIAL VALUES TO SEVERAL CONSTANTS
10280      IN=20
10290      WRITE(5,88)
10300   88 FORMAT(' ENTER INPUT FILE NAME')
10310      READ(5,89)BARIN
10320   89 FORMAT(A5)
10330      CALL IFILE(20,BARIN)
10340      CALL OFILE(21,'BAROUT')
10350      LPRT=21
10360      MONITR=5
10370      RHO=1.99
10380      GRAV=32.2
10390      PI=3.141592
10400      READ (IN,1000) NHEAD
10410      KORPOL=.FALSE.
10420      KORSTR=.FALSE.
10430      IDBT=.FALSE.
10440      NCARD=1
10450      NCASE=0
10460      NPAGE=0
10470      BNUL=0.0
10480      WFRAC=BNUL
10490      TDDUC=BNUL
```

```
      CURENT=BNUL
      GAMIN=BNUL
      HEADIN=BNUL
      CDOTIN=BNUL
      DRADIN=BNUL
      DO 10 J=1,10
      NANG(J)=0
      DO 10 I=1,11
      SANG(I,J)=BNUL
      DO 10 K=1,2
      SDAT(I,J,K)=BNUL
   10 CONTINUE
      RETURN
 1000 FORMAT (20A4)
      END
      SUBROUTINE INPUT
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
      COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
      COMMON/CMWIDE/TOWWID,BOTWID,NBARW
      COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
      COMMON/CMCHAR/TOWDRF,BOTDRF,TOWBC,BOTBC
      COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMWIND/SWIND(10),DWIND(10)
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      COMMON/CMNTIG/FIRSTP,STEP,EPS,AB,NCUTS
      COMMON/CMRUDR/ARUD(2,3),DELMAX,DLDTMX,OFSET
      COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
      COMMON/CMCNST/RHO,GRAV,PI
      COMMON/CMCOEF/A(10),B(10),AA(10),BB(10),COEF(5)
      COMMON/CMCLOG/NPOLY,NSTRIP,KORPOL,KORSTR
      COMMON/CMBOWT/BTHRUS(7),BTSPD(7),BTMAX,BTPOS,BTGAIN,NBTSPD,IDBT
      LOGICAL NPOLY,NSTRIP,KORPOL,KORSTR,IDBT
      DIMENSION NAME(9)
      DATA NAME/4HBARG,4HINIT,4HROUT,4HNTRP,4HCOEF,4HPARA,4HSTER,
     &4HTRIP,4HQUIT/
C***   THIS SUBROUTINE READS A COMPLETE SET OF INPUT DATA ON EACH PASS
    1 CONTINUE
      NCASE=NCASE+1
      NPOLY=.FALSE.
      NSTRIP=.FALSE.
      NERORS=0
   10 READ (IN,1000) LABEL,ITAG,JTAG,(TEMP(I),I=1,7)
      NCARD=NCARD+1
      JUMP=10
      DO 20 I=1,9
```

```
11000    20 IF (LABEL .EQ. NAME(I)) JUMP=I
11010       GO TO (100,200,300,400,500,600,700,800,900,30),JUMP
11020    30 CALL ERSTOP (1)
11030   100 CONTINUE
11040C***   PRESENT DATA CARD IS A BARGE/TOW DATA CARD
11050       IF (ITAG .LT. 1  .OR.  ITAG .GT. 5) CALL ERSTOP (2)
11060       GO TO (110,120,130,140,150),ITAG
11070   110 NBARW=JTAG
11080       TOWWID=TEMP(1)
11090       BOTWID=TEMP(2)
11100       GO TO 10
11110   120 NBARL=JTAG
11120       TOWLEN=TEMP(1)
11130       BOTLEN=TEMP(2)
11140       CGTOW=TEMP(3)
11150       CGBOT=TEMP(4)
11160       TOWK=TEMP(5)
11170       BOTK=TEMP(6)
11180       GO TO 10
11190   130 TOWDRF=TEMP(1)
11200       BOTDRF=TEMP(2)
11210       TOWBC=TEMP(3)
11220       BOTBC=TEMP(4)
11230       GO TO 10
11240   140 NPROP=JTAG
11250       SHP=TEMP(1)
11260       RPMAX=TEMP(2)
11270       OFSET=TEMP(3)
11280       DO 145 I=1,3
11290       ARUD(1,I)=TEMP(4)
11300   145 ARUD(2,I)=TEMP(5)
11310       DELMAX=TEMP(6)
11320       DLDTMX=TEMP(7)
11330       GO TO 10
11340   150 NBLAD=JTAG
11350       DPROP=TEMP(1)
11360       PITCH=TEMP(2)
11370       ARAT=TEMP(3)
11380       WFRAC=TEMP(4)
11390       TDDUC=TEMP(5)
11400       GO TO 10
11410   200 CONTINUE
11420C***   PRESENT DATA CARD GIVES INITIAL CONDITIONS FOR THIS CASE
11430       IF (ITAG .GT. 0) GO TO 210
11440       INSEG=JTAG
11450       SPDIN=TEMP(2)
11460       GAMIN=TEMP(4)
11470       HEADIN=TEMP(5)
11480       CDOTIN=TEMP(6)
11490       DRADIN=TEMP(7)
```

```
11500       GO TO 10
11510   210 NCUTS=JTAG
11520       STEP=TEMP(1)
11530       FIRSTP=TEMP(2)
11540       EPS=TEMP(3)
11550       AB=TEMP(4)
11560       GO TO 10
11570   300 CONTINUE
11580C***    PRESENT DATA CARD DESCRIBES CHANNEL CHARACTERISTICS
11590       IF (ITAG .GT. 0) GO TO 310
11600       NSEG=JTAG
11610       GO TO 10
11620   310 CONTINUE
11630       SANG(ITAG,JTAG)=TEMP(1)
11640       SDAT(ITAG,JTAG,1)=TEMP(2)
11650       SDAT(ITAG,JTAG,2)=TEMP(3)
11660       CUR(ITAG,JTAG,1)=TEMP(4)
11670       CUR(ITAG,JTAG,2)=TEMP(5)
11680       CUR(ITAG,JTAG,3)=TEMP(6)
11690       IF(ITAG.NE.1)GOTO 10
11700       SWIND(JTAG)=TEMP(1)*1.47
11710       DWIND(JTAG)=TEMP(7)
11720   400 CONTINUE
11730       GO TO 10
11740   500 CONTINUE
11750C***    PRESENT DATA CARD HAS HYDRODYNAMIC COEFFICIENTS
11760       IF (ITAG .LT. 1  .OR.  ITAG .GT. 5) CALL ERSTOP (2)
11770       GO TO (510,520,530,540,550), ITAG
11780   510 CONTINUE
11790       KORPOL=.TRUE.
11800       NPOLY=.TRUE.
11810       IF (NSTRIP) CALL ERSTOP (3)
11820       IF (JTAG .GT. 1) GO TO 515
11830       AA(2)=TEMP(1)
11840       AA(3)=TEMP(2)
11850       AA(9)=TEMP(3)
11860       AA(10)=TEMP(4)
11870       GO TO 10
11880   515 CONTINUE
11890       AA(1)=TEMP(1)
11900       AA(5)=TEMP(2)
11910       AA(6)=TEMP(3)
11920       AA(7)=TEMP(4)
11930       GO TO 10
11940   520 CONTINUE
11950       KORPOL=.TRUE.
11960       NPOLY=.TRUE.
11970       IF (NSTRIP) CALL ERSTOP (3)
11980       IF (JTAG .GT. 1) GO TO 525
11990       BB(2)=TEMP(1)
```

```
12000       BB(3)=TEMP(2)
12010       BB(9)=TEMP(3)
12020       BB(10)=TEMP(4)
12030       GO TO 10
12040   525 CONTINUE
12050       BB(1)=TEMP(1)
12060       BB(5)=TEMP(2)
12070       BB(6)=TEMP(3)
12080       BB(7)=TEMP(4)
12090       GO TO 10
12100   530 CONTINUE
12110       KORSTR=.TRUE.
12120       NSTRIP=.TRUE.
12130       IF (NPOLY) CALL ERSTOP (3)
12140       DO 535 I=1,5
12150   535 COEF(I)=TEMP(I)
12160       IF (JTAG .NE. 10) GO TO 10
12170       NSTRIP=.FALSE.
12180       KORPOL=.FALSE.
12190       KORSTR=.FALSE.
12200       GO TO 10
12210   540 CONTINUE
12220       IF (JTAG .GT. 0) GO TO 545
12230       IDBT=.FALSE.
12240       GO TO 10
12250   545 CONTINUE
12260       IDBT=.TRUE.
12270       NBTSPD=JTAG
12280       BTMAX=TEMP(1)
12290       BTPOS=TEMP(2)
12300       BTGAIN=TEMP(3)
12310       GO TO 10
12320   550 CONTINUE
12330       IF (JTAG .GT. 1) GO TO 555
12340       DO 551 I=1,7
12350   551 BTSPD(I)=TEMP(I)
12360       GO TO 10
12370   555 CONTINUE
12380       DO 556 I=1,7
12390   556 BTHRUS(I)=TEMP(I)
12400       GO TO 10
12410   600 CONTINUE
12420       GO TO 10
12430   700 CONTINUE
12440       BOWCLR=TEMP(1)
12450       GO TO 10
12460   800 CONTINUE
12470       KPATH=ITAG
12480       KPRINT=JTAG
12490       RETURN
```

```
  900 STOP
 1000 FORMAT (A4,1X,I1,I3,1X,7F10.5)
      END
      SUBROUTINE INITAL
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
      COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
      COMMON/CMWIDE/TOWWID,BOTWID,NBARW
      COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
      COMMON/CMCHAR/TOWDRF,BOTDRF,TOWBC,BOTBC
      COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMCNST/RHO,GRAV,PI
      COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
      COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
C***   THIS SUBROUTINE ASSIGNS INITIAL VALUES TO SEVERAL VARIABLES
      TOWDSP=TOWLEN*TOWWID*TOWDRF*TOWBC
      BOTDSP=BOTLEN*BOTWID*BOTDRF*BOTBC
      TMASS=(TOWDSP+BOTDSP)*RHO
      CGAFT=(TOWDSP*CGTOW-BOTDSP*(BOTLEN-CGBOT))/(TOWDSP+BOTDSP)
      CGFWD=TOWLEN-CGAFT
      PD=PITCH/DPROP
      ZNERTA=BOTDSP*(BOTK*BOTK+(CGAFT+BOTLEN-CGBOT)**2)
      ZNERTA=ZNERTA+TOWDSP*(TOWK*TOWK+(CGTOW-CGAFT)**2)
      ZNERTA=ZNERTA*RHO
      EFLEN=TOWLEN+BOTLEN*BOTWID/TOWWID
      EFDRF=TOWDRF
      EFBEAM=TOWWID
      TONS=(TOWDSP+BOTDSP)*RHO*GRAV/2000.0
      CALL DATOUT
      CALL HYCOEF
      DO 5 J=1,NSEG
      SANG(1,J)=0.
      NANG(J)=0
      DO 5 I=2,11
      IF(SANG(I,J).EQ.0.0)GOTO5
      SANG(1,J)=SANG(1,J)+SANG(I,J)
      NANG(J)=NANG(J)+1
    5 CONTINUE
      SPEED=SPDIN
      GAMMA=GAMIN
      HEAD=HEADIN
      CDOT=CDOTIN
      I=INSEG-1
   10 CONTINUE
```

```
13000       I=I+1
13010       IF (I .GT. NSEG) GO TO 100
13020       SEGW=SDAT(1,I,2)-SDAT(1,I,1)
13030       SEGD=(SDAT(1,I,1)+SDAT(1,I,2))/2.
13040       DRAD=DRADIN*(BOWCLR*SEGW-(SEGW-TOWWID)/2.0)
13050       RAD1=SEGD+DRAD
13060       ALDOT=SPEED/RAD1
13070       IF (SANG(1,INSEG) .GT. 0.0) GO TO 100
13080       ALDOT=-ALDOT
13090       GAMMA=GAMMA+PI
13100       HEAD=HEAD+PI
13110   100 CONTINUE
13120       CDOT=CDOT+ALDOT
13130       RETURN
13140       END
13150       SUBROUTINE ROUTCK
13160C***   ***IMPLICIT REAL*8 (A-H,O-Z)
13170       COMMON/CMPRNT/IN,LPRT,MONITR
13180       COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
13190       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
13200       COMMON/CMCNST/RHO,GRAV,PI
13210       COMMON/CMTRIP/TRPDST,ENDHED,STRDST,RSLTNT
13220C***    THIS SUBROUTINE COMPUTES THE NET DISTANCE AND HEADING CHANGE
13230C***    DURING A COMPLETE TRIP
13240       XBASE=0.0
13250       YBASE=0.0
13260       TRPANG=0.0
13270       TRPDST=0.0
13280       DO 100 J=INSEG,NSEG
13290       NSUB=NANG(J)+1
13300       DO 100 I=2,NSUB
13310       ANGRAD=SANG(I,J)/57.3
13320       SEGD=(SDAT(I,J,1)+SDAT(I,J,2))/2.
13330       TRPDST=TRPDST+SEGD*ABS(ANGRAD)
13340       SGNR=SEGD*ANGRAD/ABS(ANGRAD)
13350       XPRIM=SGNR*SIN(ANGRAD)
13360       YPRIM=SGNR*(1.0-COS(ANGRAD))
13370       XBASE=XPRIM*COS(TRPANG)-YPRIM*SIN(TRPANG)+XBASE
13380       YBASE=YPRIM*COS(TRPANG)+XPRIM*SIN(TRPANG)+YBASE
13390    30 CONTINUE
13400       TRPANG=TRPANG+ANGRAD
13410   100 CONTINUE
13420       ENDHED=TRPANG*180.0/PI
13430       STRDST=SQRT(XBASE*XBASE+YBASE*YBASE)
13440       RSLTNT=ATAN2(YBASE,XBASE)*180.0/PI
13450C***    THIS IS A TEMPORARY PRINT STATEMENT
13460       WRITE (LPRT,1000) TRPDST,ENDHED,STRDST,RSLTNT
13470 1000 FORMAT ('  CURVE,HEAD,STRAIGHT,NET=',4F15.5)
13480       RETURN
13490       END
```

```
      SUBROUTINE TRPINT
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      COMMON/CMTIME/ACUMT(100),STIME(100),NXTCRV,LSTCRV,NXTREV,LSTREV
      COMMON/CMSCR1/ALRCOM
      LOGICAL NXTCRV,LSTCRV,NXTREV,LSTREV
C***   THIS SUBROUTINE CONTROLS THE COMPUTATION OF ELAPSED TIME
C***   AS THE PROGRAM PROGRESSES ALONG A ROUTE
      TIMEL=0.0
      NSEGM=NSEG-1
      DO 100 ISEG=INSEG,NSEGM
      WRITE(LPRT,1000)ISEG
 1000 FORMAT (//'   INTEGRATION THROUGH SEGMENT NUMBER',I5//
     &' STEP  TIME     V(ANG)     ALPHA    V(HEAD)    ',
     &'  PSI    V(RAD)  RADIUS    BETA     GAMA     DELTA',
     &'   RPM    CLIN   XLIN  CLOUT  XLOUT')
   10 CONTINUE
      ALRCOM=0.0
      TSEG=0.0
      ALPHA=0.0
      NEMORY=0
      NXTCRV=.FALSE.
      LSTCRV=.FALSE.
      NXTREV=.FALSE.
      LSTREV=.FALSE.
      J=ISEG+1
      IF (J .GT. NSEG) GO TO 20
      IF (SANG(1,J)*SANG(1,ISEG) .LT. 0.0) NXTREV=.TRUE.
      NXTCRV=.TRUE.
   20 CONTINUE
      J=ISEG-1
      IF (J .LT. INSEG) GO TO 30
      IF (SANG(1,J)*SANG(1,ISEG) .LT. 0.0) LSTREV=.TRUE.
      LSTCRV=.TRUE.
   30 CONTINUE
      CALL SEGINT (TSEG,ALPHA,NEMORY)
      TIMEL=TIMEL+TSEG
      ACUMT(ISEG)=TIMEL
      STIME(ISEG)=TSEG
  100 CONTINUE
      RETURN
      END
      SUBROUTINE SEGINT (TSEG,ALPHA,NEMORY)
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
```

```
14000       COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
14010       COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
14020       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
14030       COMMON/CMCNST/RHO,GRAV,PI
14040       COMMON/CMPROP/DUM(7),NPROP,NBLAD
14050       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
14060       COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
14070      &RPMHI(3),ISEG
14080       COMMON/CMTRIP/TRPDST,ENDHED,STRDST,RSLTNT
14090       COMMON/CMTIME/ACUMT(100),STIME(100),NXTCRV,LSTCRV,NXTREV,LSTREV
14100       COMMON/CMBEGN/BEGRPM(3),BEGSPD,BEGDEL(2,3),BEGGAM,BEGHED,BEGCDT,
14110      &BEGDRD
14120       COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
14130       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
14140       COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,SIGN,CLBOW,BOWCL
14150       COMMON/SCR2/NOUTS(8),POUTS(8),NSTRS,KNT
14160       LOGICAL NXTCRV,LSTCRV,NXTREV,LSTREV,FFALSE
14170       DIMENSION Y(6)
14180C***    THIS SUBROUTINE CONTROLS THE INTEGRATION OF THE EQUATIONS OF
14190C***    MOTION THROUGH ONE SEGMENT OF A ROUTE
14200       FFALSE=.FALSE.
14210       DO 1 I=1,NPROP
14220       RPMHI(I)=RPMAX
14230       BEGDEL(1,I)=DELTA(1,I)
14240       BEGDEL(2,I)=DELTA(2,I)
14250     1 BEGRPM(I)=RPM(I)
14260       BEGSPD=SPEED
14270       BEGGAM=GAMMA
14280       BEGHED=HEAD
14290       BEGCDT=CDOT
14300       BEGDRD=DRAD
14310    10 CONTINUE
14320       NSTRS=0
14330       SIGN=SANG(1,ISEG)/ABS(SANG(1,ISEG))
14340       CALL SETY (Y,ALPHA)
14350       CALL INTEG (0,TSEG,Y,J900,FFALSE)
14360       IF (ALPHA .EQ. 0.0) NTIMES=0
14370    20 CONTINUE
14380       KNT=1
14390       NTIMES=NTIMES+1
14400       IF (ABS(Y(2)) .LT. ABS(SANG(1,ISEG)/57.3)) GO TO 30
14410       CALL TIMCOR (Y,TSEG,KPATH)
14420       GO TO 999
14430    30 CONTINUE
14440       IF((4+NTIMES)/5*5.NE.4+NTIMES)GOTO31
14450    29 CONTINUE
14460       CALL PLOTER(Y(6),Y(4),Y(2),ISEG)
14470   118 CONTINUE
14480       WRITE(5,311)
14490       READ(22,*,ERR=118)(DELTA(NR,1),NR=1,2)
```

```
14500       WRITE(5,937)(DELTA(NR,1),NR=1,2)
14510       DELTA(1,1)=DELTA(1,1)/57.3
14520       DELTA(2,1)=DELTA(2,1)/57.3
14530       DO 26 NR=2,NPROP
14540       DELTA(1,NR)=DELTA(1,1)
14550    26 DELTA(2,NR)=DELTA(2,1)
14560   119 WRITE(5,313)
14570       READ(22,*,ERR=119)(RPMHI(NR),NR=1,NPROP)
14580       WRITE(5,937)(RPMHI(NR),NR=1,NPROP)
14590   937 FORMAT(2F5.0)
14600       IF(ABS(DELTA(1,1)).GT.1.57)GOTO32
14610       CALL SETIC (Y,TSEG,ALPHA,NTIMES,NEMORY)
14620    31 CONTINUE
14630       CALL INTEG (6,TSEG,Y,J900,FFALSE)
14640   311 FORMAT(' RUDDER=')
14650   312 FORMAT(F10.5)
14660   313 FORMAT(' THROTTLE=')
14670       ALPHA=Y(2)
14680       GOTO33
14690    32 CONTINUE
14700       CALL SETIC (Y,TSEG,ALPHA,NTIMES,NEMORY)
14710    33 CONTINUE
14720       GAMA=ATAN2(Y(5),Y(6)*Y(1))
14730       IF (SIGN .LT. 0.0) GAMA=GAMA+PI
14740       BETA=Y(4)-GAMA
14750       RPMAV=(RPM(1)+RPM(2))/2.
14760       DELTAS=(DELTA(1,1)+DELTA(1,2))/2.
14770       WRITE (LPRT,2000) NTIMES,TSEG,(Y(I),I=1,6),BETA,GAMA,
14780      &DELTAS,RPMAV,CLIN,XLIN,CLOUT,XLOUT
14790       IF(ABS(DELTA(1,1)).GT.1.57)GOTO29
14800       IF (NTIMES .LT. 200) GO TO 20
14810       GO TO (999,910,35),KPATH
14820    35 CONTINUE
14830       TSEG=-1.0
14840       GO TO 950
14850   900 WRITE (LPRT,1000)
14860       GO TO 999
14870   910 CONTINUE
14880       TSEG=0.0
14890       CALL TIMCOR (Y,TSEG,4)
14900   950 WRITE (LPRT,3000)
14910   999 RETURN
14920  1000 FORMAT ('  ITERATION FAILURE')
14930  2000 FORMAT (I5,F7.0,E12.3,F8.3,E12.3,2F9.4,F8.0,3F9.4,5F7.0)
14940  3000 FORMAT ('  PRECEEDING SEGMENT NOT COMPLETED WITHIN ALLOWED LIMITS'
14950      &)
14960       END
14970       SUBROUTINE ERSTOP (NGOOF)
14980C***   ***IMPLICIT REAL*8 (A-H,O-Z)
14990       COMMON/CMPRNT/IN,LPRT,MONITR
```

```
15000       COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
15010       COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
15020C***    THIS SUBROUTINE WRITES ERROR MESSAGES
15030       NERORS=NERORS+1
15040       GO TO (10,20,30),NGOOF
15050    10 WRITE (LPRT,1001) NCARD,LABEL
15060       GO TO 999
15070    20 WRITE (LPRT,1002) NCARD,LABEL,ITAG,JTAG,ITAG
15080       GO TO 999
15090    30 WRITE (LPRT,1003) NCARD,LABEL,ITAG,JTAG,LABEL
15100       GO TO 999
15110   999 RETURN
15120  1001 FORMAT (' IN CARD NO.',I5,'  OF CURRENT DATA FILE, THE LABEL
15130      &A4,'  CAN NOT BE INTERPRETED')
15140  1002 FORMAT (' IN CARD NO.',I5,'  OF CURRENT DATA FILE, LABELLED
15150      &A4,1X,I1,I3/' THE VALUE OF ITAG,  ',I1,
15160      &'  IS NOT IN THE PERMISSABLE RANGE')
15170  1003 FORMAT (' CARD NO.',I5,'  OF CURRENT DATA FILE, LABELLED
15180      &A4,1X,I1,I3/' IS INCONSISTENT WITH A PREVIOUS  ',
15190      &A4,'  CARD')
15200  9999 RETURN
15210       END
15220       SUBROUTINE INTEG (NEQ,X,Y,ISTAR,STPSZ)
15230C***   ***IMPLICIT REAL*8 (A-H,O-Z)
15240       COMMON/CMPRNT/IN,LPRT,MONITR
15250       COMMON/CMNTIG/FIRSTP,STEP,EPS,AB,NCUTS
15260       INTEGER NEQ, NCUTS
15270C***   *** REAL*8 X,STEP,Y(6),EPS,AB
15280       LOGICAL STPSZ
15290C***   *** REAL*8 HC/0.0D0/,FINAL,H2,H3,H6,H8,ERR,TEST,T,H,EPSL,TEMPO
15300       DIMENSION Y(6),Y1(6),Y2(6),F0(6),F1(6),F2(6)
15310       INTEGER CUT
15320       LOGICAL DBL
15330       DATA HC/0.0/
15340C***   ***      REAL*8 Y1(6),Y2(6),F0(6),F1(6),F2(6)
15350    50 FORMAT (' THE STEPSIZE IS NOW',1PD15.6, ' AT TAU =',D15.6)
15360    60 FORMAT ('    THE STEPSIZE HAS BEEN HALVED ',I3,' TIMES')
15370       IF(NEQ.NE.0)  GO TO 10
15380       HC = FIRSTP
15390       RETURN
15400    10 IF(STEP.EQ.0)  RETURN
15410       IF(HC.EQ.0) HC = STEP
15420       FINAL = X+STEP
15430       H = STEP
15440       EPSL = EPS
15450       IF(EPS.EQ.0 .OR.ABS(H) .LE.ABS(HC)) GO TO 15
15460       IF(H*HC.LE.0)  HC = -HC
15470       H = HC
15480    15 T = X+H
15490       CUT = NCUTS
```

```
15500       X = FINAL
15510       H2 = H/2.
15520       H3 = H/3.
15530       H6 = H/6.
15540       H8 = H/8.
15550    20 IF(H.GT.0 .AND. T.GT.FINAL .OR. H.LT.0.AND.T.LT.FINAL) GOTO 40
15560    21 CALL FORCE(T-H,Y,F0)
15570       DO 22 I = 1,NEQ
15580    22 Y1(I) = F0(I)*H3+Y(I)
15590       CALL FORCE(T-2.*H3,Y1,F1)
15600       DO 23 I = 1,NEQ
15610    23 Y1(I) = (F0(I)+F1(I))*H6+Y(I)
15620       CALL FORCE(T-2.*H3,Y1,F1)
15630       DO 24 I = 1,NEQ
15640    24 Y1(I) = (F1(I)*3.+F0(I))*H8+Y(I)
15650       CALL FORCE(T-H2,Y1,F2)
15660       DO 25 I = 1,NEQ
15670    25 Y1(I) = (F2(I)*4.-F1(I)*3.+F0(I))*H2 +Y(I)
15680       CALL FORCE(T,Y1,F1)
15690       DO 26 I = 1,NEQ
15700    26 Y2(I) = (F2(I)*4.+F1(I)+F0(I))*H6 +Y(I)
15710       IF(EPSL.EQ.0) GO TO 38
15720       DBL = .TRUE.
15730       DO 35 I = 1,NEQ
15740       ERR =ABS(Y1(I)-Y2(I))*0.2
15750       TEST =ABS(Y1(I))*EPSL
15760       IF(ERR.LE.TEST .OR.  ERR.LT.AB) GO TO 34
15770       H = H2
15780       T = T-H2
15790       IF (.NOT.STPSZ) GO TO 30
15800       TEMPO = T-H2
15810       WRITE (LPRT,50) H, TEMPO
15820    30 CUT = CUT - 1
15830       IF (CUT .GE. 0) GO TO 31
15840       X = T - H2
15850       WRITE (LPRT,60) NCUTS
15860       RETURN
15870    31 IF(T+H.NE.T) GO TO 33
15880       X = T
15890       RETURN
15900    33 H2 = H/2.
15910       H3 = H/3.
15920       H6= H/6.
15930       H8 = H/8.
15940       GO TO 21
15950    34 IF(64.0*ERR.GT.TEST)  DBL = .FALSE.
15960    35 CONTINUE
15970       IF(.NOT.DBL) GO TO 38
15980       H2 = H
15990       H = 2.*H
```

```
16000       IF (STPSZ)      WRITE (LPRT,50) H,T
16010       H3 = H/3.
16020       H6 = H/6.
16030       H8 = H/8.
16040       CUT = NCUTS
16050    38 DO 39 I = 1,NEQ
16060    39 Y(I) = Y2(I)
16070       T = T+H
16080       GO TO 20
16090    40 IF(EPSL.EQ.0)  RETURN
16100       HC = H
16110       H = FINAL-(T-H)
16120       IF(ABS(H).LE.ABS(FINAL)*9.536744D-7) RETURN
16130       T= FINAL
16140       EPSL = 0
16150       H2 = H/2.
16160       H3 = H/3.
16170       H6 = H/6.
16180       H8 = H/8.
16190       GO TO 20
16200       END
16210       SUBROUTINE FORCE (TM,Y,DY)
16220C***   ***IMPLICIT REAL*8 (A-H,O-Z)
16230       COMMON/CMPRNT/IN,LPRT,MONITR
16240       COMMON/CMWIDE/TOWWID,BOTWID,NBARW
16250       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
16260       COMMON/CMCNST/RHO,GRAV,PI
16270       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
16280       COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
16290       COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
16300      &RPMHI(3),ISEG
16310       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
16320       COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,XSIGN,CLBOW,BOWCL
16330       COMMON/CMCOEF/A(10),B(10),AA(10),BB(10),COEF(5)
16340       COMMON/CMCLOG/NPOLY,NSTRIP,KORPOL,KORSTR
16350       COMMON/CMSCR1/ALRCOM
16360       DIMENSION Y(6),DY(6)
16370       LOGICAL JUMP
16380       EXTERNAL SIGN
16390C***    THIS SUBROUTINE COMPUTES HYDRODYNAMIC FORCES
16400       DY(2)=Y(1)
16410       DY(4)=Y(3)
16420       DY(6)=Y(5)
16430       CALL CURRNT(Y(2),Y(4),Y(6),CTAN,CRAD,CROT)
16440       CY=Y(5)-CRAD
16450       CX=Y(6)*Y(1)
16460       SPD=SQRT(CY*CY+(CX*XSIGN+CTAN)**2)
16470       APGAMA=ATAN2(CY,(CX+CTAN*XSIGN))
16480       IF (XSIGN .LT. 0.0) APGAMA=APGAMA+PI
16490       APBETA=Y(4)-APGAMA
```

```
16500       VADV=SPD*COS(APBETA)
16510       ANGRAT=Y(3)-Y(1)-CROT
16520       CALL WINDFO (Y(2),Y(4),FWLON,FWLAT,FWROT)
16530       CALL THRUST (VADV,PUSH,TORQ)
16540       CALL RESIST (VADV,DRAG)
16550       CALL RUDFOR (VADV,PROMOM,DELTA,FRUD,DRUD)
16560       CALL BOWTHR (VADV,BTPUSH,BTSWAY,BTMOM)
16570       JUMP=.FALSE.
16580       CALL HYDRO (ANGRAT,SPD,APBETA,SMOM,FSWAY)
16590       FPROP=PUSH-DRAG+BTPUSH+FWLON
16600    10 CONTINUE
16610       FSIDE=FRUD+FSWAY+BTSWAY+FWLAT
16620       QALPH=FPROP*COS(Y(4))-FSIDE*SIN(Y(4))
16630       QALPH=QALPH*Y(6)
16640       QRAD=FPROP*SIN(Y(4))+FSIDE*COS(Y(4))
16650       QHEAD=SMOM-FRUD*(CGAFT+BOTLEN)+BTMOM+PROMOM+FWROT
16660       RRM=TMASS*Y(6)*Y(6)
16670       QKEEP=QALPH/RRM+QHEAD/RRM
16680       DY(1)=QKEEP-2.0*Y(5)*Y(1)/Y(6)
16690       DY(3)=QKEEP+QHEAD/ZNERTA-2.0*Y(5)*Y(1)/Y(6)
16700       DY(5)=QRAD/TMASS+Y(6)*Y(1)*Y(1)
16710       IF (JUMP) GO TO 100
16720       ACC=SQRT(DY(1)*DY(1)*Y(6)*Y(6)+DY(5)*DY(5))
16730       GA=ATAN2(DY(5),Y(6)*DY(1))
16740       BA=Y(4)-GA
16750       VDOT=-ACC*SIN(BA)
16760       FSWAY=FSWAY+B(10)*TLEN*TLEN*TLEN*VDOT*RHO/2.0
16770       SMOM=SMOM+A(9)*TLEN*TLEN*TLEN*TLEN*(DY(3)-DY(1))*RHO/2.0
16780       JUMP=.TRUE.
16790       GO TO 10
16800   100 CONTINUE
16810C***    WRITE (MONITR,1) DRAG,PUSH,FPROP,FSWAY,FRUD,FSIDE
16820C***    WRITE (MONITR,1) QALPH,QRAD,QHEAD,SPD,APGAMA
16830C***    WRITE (MONITR,1) (DY(I),I=1,6)
16840C***  1 FORMAT (' CHECK',6E15.5)
16850       RETURN
16860       END
16870       SUBROUTINE PLOTER(RADIUS,PSI,ALPHA,I)
16880C***   ***IMPLICIT REAL*8 (A-H,O-Z)
16890       COMMON/CMPRNT/IN,LPRT,MONITR
16900       COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
16910       COMMON/CMWIDE/TOWWID,BOTWID,NBARW
16920       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
16930       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
16940       COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
16950      &RPMHI(3),ISEG
16960       COMMON/CMCNST/RHO,GRAV,PI
16970       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
16980       COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,XSIGN,CLBOW,BOWCL
16990       COMMON/CMSYMB/ROW(80),SYM(4),BL,NMAX
```

```
17000       COMMON/SCR2/NOUTS(8),POUTS(8),NSTRS,KNT
17010       DATASYM,BL/1HI,1HO,1HB,1HT,1H /
17020       DATANOUTS/5HSPEED,5HRPM-P,5HRPM-S,5H  -CL,5HV-RAD,5HV-YAW,
17030      &5HS-RUD,5HF-RUD/
17040       DIST(X,Y)=SQRT(X*X+Y*Y+2.0*X*Y*SIN(PSI))
17050       RPRIM(Z)=RADIUS+Z/COS(PSI)
17060       XLEN(V,W)=V-W*SIN(PSI)/COS(PSI)
17070       EXES(T,U)=XSIGN*(RADIUS*ALPHA+T*COS(PSI)-U)
17080       NSTRS=NSTRS+1
17090       IF((NSTRS+1)/2*2.EQ.(NSTRS+1))WRITE(MONITR,96)NHEAD
17100    96 FORMAT('1',20A4)
17110       XSIGN=SANG(1,I)/ABS(SANG(1,I))
17120       TWID=TOWWID/2.0*XSIGN
17130       BWID=BOTWID/2.0*XSIGN
17140       PCRT=ALPHA/SANG(1,I)*5730.
17150       WRITE(MONITR,97)PCRT,I
17160    97 FORMAT(//F5.1,' PERCENT THRU SEGMENT',I3)
17170       XINC=(TOWLEN+BOTLEN)/7.
17180       POUTS(1)=SPEED/1.47
17190       POUTS(2)=RPM(1)
17200       POUTS(3)=RPM(2)
17210       POUTS(4)=RPM(3)
17220       POUTS(5)=SPEED*SIN(GAMMA)/1.47
17230       POUTS(6)=CDOT*57.3
17240       POUTS(7)=DELTA(1,1)*57.3
17250       CLIN=9999.E+10
17260       POUTS(8)=DELTA(2,1)*57.3
17270       CLOUT=9999.E+10
17280       DO102JP=1,25
17290       NMAX=0
17300       DO10K=1,80
17310    10 ROW(K)=BL
17320       XPOS=XINC*FLOAT(20-JP)
17330       CALL BARDIS(ALPHA,RADIUS,PSI,XPOS,DPERI,DPERO,
17340      &PERRAD,PERTAN,ISSEGO,ISEGO)
17350       IF(DPERI.GT.CLIN)GOTO12
17360       CLIN=DPERI
17370       XLIN=XPOS
17380    12 IF(DPERO.GT.CLOUT)GOTO14
17390       CLOUT=DPERO
17400       XLOUT=XPOS
17410    14 CONTINUE
17420       CALL SCALE(-DPERI,I,1)
17430       CALL SCALE(DPERO,I,2)
17440       IF(XPOS.GT.CGFWD)GOTO100
17450       IF(XPOS.LT.-CGAFT-BOTLEN)GOTO100
17460       WWID=TWID
17470       NSYM=3
17480       IF(XPOS.GT.-CGAFT)GOTO20
17490       WWID=BWID
```

```
17500       NSYM=4
17510    20 CONTINUE
17520       CL=-WWID/COS(PSI)
17530       CALL SCALE(CL,I,NSYM)
17540       CL=WWID/COS(PSI)
17550       CALL SCALE(CL,I,NSYM)
17560   100 JK=JP-14
17570       IF(JK.LT.1.OR.JK.GT.8)GOTO101
17580       WRITE(MONITR,98)NOUTS(JK),POUTS(JK),(ROW(JJ),JJ=11,NMAX)
17590    98 FORMAT(1X,A5,F5.1,70A1)
17600       GOTO102
17610   101 WRITE(MONITR,99)(ROW(JJ),JJ=11,NMAX)
17620   102 CONTINUE
17630    99 FORMAT(11X,70A1)
17640       RETURN
17650       END
17660       SUBROUTINE SCALE(CL,I,ICHAR)
17670C***   ***IMPLICIT REAL*8 (A-H,O-Z)
17680       COMMON/CMPRNT/IN,LPRT,MONITR
17690       COMMON/CMWIDE/TOWWID,BOTWID,NBARW
17700       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
17710       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
17720       COMMON/CMCNST/RHO,GRAV,PI
17730       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
17740       COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,XSIGN,CLBOW,BOWCL
17750       COMMON/CMSYMB/ROW(80),SYM(4),BL,NMAX
17760C***   TEMPORARY FIX FOR SEG WIDTH
17770       SEGW=SDAT(1,I,2)-SDAT(1,I,1)
17780       CL=XSIGN*CL
17790       N=IFIX(CL/SEGW*35.)+45
17800       IF(N.LT.11)GOTO10
17810       IF(N.GT.80)GOTO10
17820       ROW(N)=SYM(ICHAR)
17830       IF(N.GT.NMAX)NMAX=N
17840    10 RETURN
17850       END
17860       SUBROUTINE THRUST (VADV,PUSH,TORQ)
17870C***   ***IMPLICIT REAL*8 (A-H,O-Z)
17880       COMMON/CMPRNT/IN,LPRT,MONITR
17890       COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
17900       COMMON/CMCNST/RHO,GRAV,PI
17910       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
17920       COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
17930      &RPMHI(3),ISEG
17940       COMMON/SCR4/PUSHEA(3)
17950       LOGICAL LIMIT,NIMIT
17960C***    THIS SUBROUTINE COMPUTES THE THRUST AND TORQUE AT THE PRESEN
17970C***    PROPELLER RPM AND ADJUSTS THE RPM TO THE MAXIMUM PERMITTED B
17980C***    EITHER COMMAND OR HORSEPOWER LIMITATIONS
17990       PUSH=0.0
```

```
18000       DO 200 IP=1,NPROP
18010       RPM(IP)=RPMHI(IP)
18020       LIMIT=.FALSE.
18030       NIMIT=.FALSE.
18040       LOOP=0
18050       XSIN=SIGN(1.0,RPMHI(IP))
18060     5 CONTINUE
18070       RPS=RPM(IP)/60.
18080       LOOP=LOOP+1
18090       IF (LOOP .GT. 100) GO TO 50
18100       IF (NBLAD .LT. 0) GO TO 20
18110       CALL QUAD4(VADV,RPS,PUSHEA(IP),TORQ)
18120       GOTO 30
18130    20 CONTINUE
18140       CJ=(1.0-WFRAC)*VADV/RPS/DPROP
18150       CALL WAGN4 (CJ,TKT,QKQ)
18160       RN2D4=RHO*RPS*RPS*DPROP**4
18170       PUSHEA(IP)=TKT*RN2D4*(1.0-TDDUC)
18180       TORQ=QKQ*RN2D4*DPROP
18190    30 CONTINUE
18200       HP=PI*ABS(RPS*TORQ)/275.0
18210       IF (HP .GT. SHP  .OR.  ABS(RPM(IP)) .GT. ABS(RPMHI(IP))) GO TO 40
18220       NIMIT=.TRUE.
18230       IF (LIMIT) GO TO 50
18240       RPM(IP)=RPM(IP)+5.*XSIN
18250       IF (ABS(RPM(IP)) .LT. ABS(RPMHI(IP))+3.) GO TO 5
18260       RPM(IP)=RPM(IP)-5.*XSIN
18270       GO TO 50
18280    40 CONTINUE
18290       IF(NIMIT)GO TO 42
18300       GO TO 46
18310    42 IF (.NOT. LIMIT) GO TO 45
18320       GO TO 46
18330    45 NIMIT=.FALSE.
18340    46 LIMIT=.TRUE.
18350       IF (NIMIT) GO TO 50
18360       RPM(IP)=RPM(IP)-5.*XSIN
18370       GOTO 5
18380    50 CONTINUE
18390C***         WRITE (LPRT,100) VADV,RPS,DPROP,CJ
18400C***         WRITE (LPRT,100) RPM(IP),RN2D4,TKT,QKQ
18410C***         WRITE (LPRT,100) PUSH,TORQ,HP,SHP
18420C***   100 FORMAT (' PROP',4E15.5)
18430   200 PUSH=PUSH+PUSHEA(IP)
18440       RETURN
18450       END
18460       SUBROUTINE RESIST (VADV,DRAG)
18470C***   ***IMPLICIT REAL*8 (A-H,O-Z)
18480       COMMON/CMPRNT/IN,LPRT,MONITR
18490       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
```

```
18500       EXTERNAL SIGN
18510C***    THIS SUBROUTINE COMPUTES DEEP WATER, UNRESTRICTED CHANNEL RE
18520C***    RESISTANCE
18530C***    FOR THE BARGE AND TOWBOAT
18540       EL2=SQRT(EFLEN)
18550       IF (EL2 .GT. 20.0) GO TO 10
18560       CDRAG=1.723578*EL2
18570       CDRAG=CDRAG+5.520842E-3*EL2**3
18580       CDRAG=CDRAG-2.542E-6*EL2**5
18590       GO TO 20
18600    10 CONTINUE
18610       CDRAG=2.63833*EL2
18620       CDRAG=CDRAG+2.21667E-3*EL2**3
18630    20 CONTINUE
18640       DRAG=184.0*(TONS/CDRAG)**2.86
18650       DRAG=DRAG*(ABS(VADV)/EFBEAM)**1.86
18660       DRAG=DRAG*SIGN(1.0,VADV)/EFDRF**2.49
18670       RETURN
18680       END
18690       SUBROUTINE WAGN4 (CJ,TKT,QKQ)
18700C***   ***IMPLICIT REAL*8 (A-H,O-Z)
18710       COMMON/CMPRNT/IN,LPRT,MONITR
18720       COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
18730C***    COMPUTE THRUST COEFFICIENT
18740       TKT=-0.719975E-2
18750       TKT=TKT-0.790916E-1*ARAT
18760       TKT=TKT-0.179541   *CJ
18770       TKT=TKT-0.625748E-1*ARAT*CJ
18780       TKT=TKT-0.311639   *CJ*CJ
18790       TKT=TKT+0.143160   *ARAT*ARAT*CJ*CJ*CJ
18800       TKT=TKT+0.531326   *PD
18810       TKT=TKT-0.114389   *ARAT*PD*CJ
18820       TKT=TKT+0.625376E-1*PD*CJ*CJ
18830       TKT=TKT+0.125537   *PD*CJ*CJ*CJ
18840       TKT=TKT-0.523821E-1*ARAT*PD*CJ*CJ*CJ
18850       TKT=TKT-0.207108   *PD*PD
18860       TKT=TKT+0.270781   *ARAT*PD*PD
18870       TKT=TKT+0.134182   *PD*PD*CJ
18880       TKT=TKT-0.121086   *ARAT*PD*PD*CJ
18890       TKT=TKT-0.189764E-1*ARAT*ARAT*ARAT*PD*PD*CJ
18900       TKT=TKT-0.439535E-1*ARAT*ARAT*ARAT*PD*PD*CJ*CJ
18910       TKT=TKT-0.624937E-1*PD*PD*CJ*CJ*CJ
18920       TKT=TKT-0.496939E-2*ARAT*ARAT*PD**6
18930       TKT=TKT+0.115986E-1*ARAT*ARAT*PD**6*CJ
18940C***    COMPUTE TORQUE COEFFICIENT
18950       QKQ=0.964375E-2
18960       QKQ=QKQ-0.104103E-1*ARAT
18970       QKQ=QKQ+0.512431E-2*ARAT*ARAT
18980       QKQ=QKQ+0.109936E-1*ARAT*ARAT*ARAT
18990       QKQ=QKQ-0.453419E-2*CJ
```

```
19000       QKQ=QKQ+0.216078E-1*ARAT*CJ
19010       QKQ=QKQ-0.507337E-1*CJ*CJ
19020       QKQ=QKQ+0.377970E-1*ARAT*CJ*CJ
19030       QKQ=QKQ-0.549486E-1*ARAT*ARAT*ARAT*CJ*CJ*CJ
19040       QKQ=QKQ-0.507319E-1*ARAT*ARAT*PD
19050       QKQ=QKQ+0.368649E-1*PD*CJ
19060       QKQ=QKQ-0.106520    *ARAT*PD*CJ
19070       QKQ=QKQ+0.465315E-1*ARAT*ARAT*ARAT*PD*CJ*CJ
19080       QKQ=QKQ+0.883010E-1*ARAT*ARAT*PD*CJ*CJ*CJ
19090       QKQ=QKQ+0.112619E-1*PD*PD
19100       QKQ=QKQ+0.104825    *ARAT*PD*PD
19110       QKQ=QKQ-0.449154E-1*ARAT*PD*PD*CJ
19120       QKQ=QKQ+0.378780E-1*ARAT*ARAT*PD*PD*CJ
19130       QKQ=QKQ+0.177304E-1*PD*PD*CJ*CJ
19140       QKQ=QKQ-0.164687E-1*ARAT*PD*PD*CJ*CJ
19150       QKQ=QKQ-0.344328E-1*ARAT*ARAT*PD*PD*CJ*CJ
19160       QKQ=QKQ-0.249132E-1*ARAT*ARAT*ARAT*PD*PD*CJ*CJ
19170       QKQ=QKQ-0.233007E-1*ARAT*PD*PD*CJ*CJ*CJ
19180       QKQ=QKQ-0.120209E-2*PD**6
19190       QKQ=QKQ-0.118997E-2*ARAT*ARAT*ARAT*PD**6
19200       QKQ=QKQ+0.458094E-2*ARAT*PD**6*CJ
19210       RETURN
19220       END
19230       SUBROUTINE QUAD4(VADV,RPS,PUSH1,TORQ)
19240C***   ***IMPLICIT REAL*8 (A-H,O-Z)
19250       COMMON/CMPRNT/IN,LPRT,MONITR
19260       COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
19270       COMMON/CMCNST/RHO,GRAV,PI
19280       DIMENSION AT(21),BT(21),AQ(21),BQ(21)
19290       DATA AT/0.25350E-1,0.17820E+0,0.14674E-1,
19300      &0.28054E-1,-.16328E-1,-.53041E-1,
19310      &0.60605E-3,0.36823E-1,-.25429E-2,
19320      &-.17680E-1,0.27331E-2,0.21436E-1,
19330      &-.24782E-2,0.12317E-2,0.50980E-2,
19340      &0.78076E-2,-.37816E-2,0.35353E-2,
19350      &0.53014E-2,0.21940E-2,-.28306E-2/
19360       DATA BT/0.00000E+0,-.74777E+0,-.13822E-1,
19370      &0.10077E+0,-.11318E-1,0.47186E-1,
19380      &0.10666E-1,-.90239E-2,-.78452E-2,
19390      &0.23941E-1,0.80787E-2,-.14942E-3,
19400      &-.31925E-2,0.92620E-2,0.15527E-2,
19410      &-.65683E-2,-.61655E-3,0.51033E-2,
19420      &-.60263E-3,-.82244E-2,-.63789E-3/
19430       DATA AQ/0.24645E-1,0.26718E+0,0.16056E-1,
19440      &0.65822E-1,-.22497E-1,-.78062E-1,
19450      &0.24126E-2,0.61475E-1,-.16065E-1,
19460      &-.33291E-1,0.12311E-1,0.31123E-1,
19470      &-.12559E-1,0.13948E-1,0.88397E-2,
19480      &0.50358E-3,-.79990E-2,0.13345E-1,
19490      &0.11928E-1,-.13556E-2,-.70825E-2/
```

```
      DATA BQ/0.00000E+0,-.11081E+1,0.15909E-2,
     &0.16455E+0,-.20601E-1,0.85343E-1,
     &0.87856E-2,-.31327E-1,-.96650E-2,
     &0.43190E-1,0.12453E-1,0.95986E-3,
     &-.79986E-2,0.15073E-1,0.24595E-2,
     &-.16918E-1,0.51603E-2,0.11504E-1,
     &-.47976E-2,-.14566E-1,0.23280E-2/
      TKT=0.0
      QKQ=0.0
      WF=WFRAC
      IF(VADV.LT.0.)WF=0.0
      PFAC=0.7*PI*RPS*DPROP
      CJB=ATAN2((1.0-WF)*VADV,PFAC)
      DO 10 K=1,21
      XK=FLOAT(K-1)*CJB
      TKT=TKT+AT(K)*COS(XK)+BT(K)*SIN(XK)
   10 QKQ=QKQ+AQ(K)*COS(XK)+BQ(K)*SIN(XK)
      PUSH1=0.125*RHO*(VADV*VADV+PFAC*PFAC)*PI*DPROP*DPROP
      TORQ=PUSH1*QKQ*DPROP/10.
      PUSH1=TKT*PUSH1
      RETURN
      END
      SUBROUTINE TIMCOR (Y,TSEG,KSTOP)
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMWIDE/TOWWID,BOTWID,NBARW
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMCNST/RHO,GRAV,PI
      COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
      DIMENSION Y(6)
C***   THIS SUBROUTINE COMPUTES THE ELAPSED TIME CORRECTION REQUIRE
C***   WHEN THE INTEGRATION OVERSHOOTS THE END OF A SEGMENT
C***   VARIABLES DESCRIBING THE VELOCITY AND ORIENTATION OF THE BAR
C***   ARE INITIALIZED FOR THE START OF THE NEXT SEGMENT
      IF (KSTOP .EQ. 4) GO TO 2
      SPEED=SQRT(Y(1)*Y(1)*Y(6)*Y(6)+Y(5)*Y(5))
      GAMMA=ATAN2(Y(5),Y(6)*Y(1))
      IF (SANG(1,ISEG) .LT. 0.0) GAMMA=GAMMA+PI
      SEGD=(SDAT(1,ISEG,1)+SDAT(1,ISEG,2))/2.
      HEAD=Y(4)
      DRAD=Y(6)-SEGD
      CDOT=Y(3)
      IF (KSTOP .EQ. 3) GO TO 999
      TCOR=Y(6)*(ABS(Y(2))-ABS(SANG(1,ISEG)/57.3))/SPEED
      HEAD=Y(4)-TCOR*Y(3)
      DRAD=Y(6)-TCOR*Y(5)-SEGD
```

```
20000       I=ISEG+1
20010       IF (I .GT. NSEG) GO TO 30
20020     2 CONTINUE
20030       CHSIGN=1.0
20040       J=ISEG
20050     5 CONTINUE
20060       J=J+1
20070       IF (J .GT. NSEG) GO TO 30
20080       SEGW=SDAT(1,J,2)-SDAT(1,J,1)
20090       IF (SANG(1,INSEG)*SANG(1,J) .LT. 0.0) CHSIGN=-1.0
20100       HEAD=HEADIN*CHSIGN
20110       GAMMA=GAMIN*CHSIGN
20120       CDOT=CDOTIN*CHSIGN
20130       DRAD=DRADIN*(BOWCLR*SEGW-(SEGW-TOWWID)/2.0)
20140       IF (SANG(1,J) .GT. 0.0) GO TO 30
20150       GO TO 15
20160    10 CONTINUE
20170       IF (SANG(1,ISEG)*SANG(1,I) .GT. 0.0) GO TO 30
20180       DRAD=-DRAD
20190       IF (SANG(1,I) .GT. 0.0) GO TO 20
20200    15 CONTINUE
20210       HEAD=HEAD+PI
20220       GAMMA=GAMMA+PI
20230       GO TO 30
20240    20 CONTINUE
20250       HEAD=HEAD-PI
20260       GAMMA=GAMMA-PI
20270    30 CONTINUE
20280       TSEG=TSEG-TCOR
20290       WRITE (LPRT,100) SPEED,TCOR,CDOT,HEAD,GAMMA,DRAD
20300   100 FORMAT ('***S,T,C,H,G,D***',6F15.5)
20310   999 RETURN
20320       END
20330       SUBROUTINE RUDFOR(VADV,PROMOM,DELTA,FRUD,DRUD)
20340C***   ***IMPLICIT REAL*8 (A-H,O-Z)
20350       COMMON/CMPRNT/IN,LPRT,MONITR
20360       COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
20370       COMMON/CMCNST/RHO,GRAV,PI
20380       COMMON/CMRUDR/ARUD(2,3),DELMAX,DLDTMX,OFSET
20390       COMMON/SCR4/PUSHEA(3)
20400       DIMENSIONDELTA(2,3)
20410C***    THIS SUBROUTINE COMPUTES RUDDER LIFT AND DRAG FORCES
20420       FRUD=0.0
20430       DRUD=0.0
20440       DO 100 IS=1,2
20450       DO 100 IT=1,NPROP
20460       CDRUD=0.0
20470       CDRUD=CDRUD+0.0*DELTA(IS,IT)
20480       CDRUD=CDRUD+0.0*DELTA(IS,IT)*DELTA(IS,IT)
20490       CDRUD=CDRUD+0.0*DELTA(IS,IT)*DELTA(IS,IT)*DELTA(IS,IT)
```

```
      CLRUD=0.0
      CLRUD=CLRUD+2.0*DELTA(IS,IT)
      CLRUD=CLRUD+0.0*DELTA(IS,IT)*DELTA(IS,IT)
      CLRUD=CLRUD+0.0*DELTA(IS,IT)*DELTA(IS,IT)*DELTA(IS,IT)
C***  TEMP STMNT TO COMPUTE AVG PUSH
      APROP=PI*DPROP*DPROP/4.0
      FACTOR=PUSHEA(IT)/APROP+VADV*ABS(VADV)*RHO/2.0
      FRUD=CLRUD*ARUD(IS,IT)*FACTOR+FRUD
      DRUD=CDRUD*ARUD(IS,IT)*FACTOR+DRUD
  100 CONTINUE
      PROMOM=(PUSHEA(1)-PUSHEA(2))*OFSET
      RETURN
      END
      SUBROUTINE DATOUT
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
      COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
      COMMON/CMWIDE/TOWWID,BOTWID,NBARW
      COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
      COMMON/CMCHAR/TOWDRF,BOTDRF,TOWBC,BOTBC
      COMMON/CMPROP/SHP,DPROP,PITCH,ARAT,WFRAC,TDDUC,PD,NPROP,NBLAD
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      COMMON/CMNTIG/FIRSTP,STEP,EPS,AB,NCUTS
      COMMON/CMRUDR/ARUD(2,3),DELMAX,DLDTMX,OFSET
      COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
      COMMON/CMBOWT/BTHRUS(7),BTSPD(7),BTMAX,BTPOS,BTGAIN,NBTSPD,IDBT
      LOGICAL IDBT
C***   THIS SUBROUTINE PRINTS OUT THE INPUT DATA DESCRIBING THE BAR
C***   AND TOWBOAT CHARACTERISTICS
      WRITE (LPRT,9000)
      WRITE (LPRT,1000) NHEAD,NBARL,NBARW
      WRITE (LPRT,2000) TOWLEN,BOTLEN,TOWWID,BOTWID,TOWDRF,BOTDRF
      WRITE (LPRT,3¬00) TOWBC,BOTBC,CGTOW,CGBOT,TOWK,BOTK
      WRITE (LPRT,4000) NPROP,SHP,RPMAX,OFSET
      WRITE (LPRT,5000) NBLAD,DPROP,PITCH,ARAT,WFRAC,TDDUC
      WRITE (LPRT,6000) ARUD(1,1),ARUD(2,1),DELMAX,DLDTMX
      WRITE (LPRT,7000) BOWCLR,SPDIN,GAMIN,HEADIN,CDOTIN,DRADIN
      WRITE (LPRT,8000) NCUTS,FIRSTP,STEP,EPS,AB
      WRITE (LPRT,9000)
 1000 FORMAT (20X,20A4////10X,'CHARACTERISTICS OF TOWBOAT AND',I3,
     &'  LONG BY',I3,'  WIDE BARGE TOW'///37X,'TOW',11X,'BOAT'/)
 2000 FORMAT (15X,'LENGTH OVERALL',F12.0,F15.0/15X,'WIDTH',F21.0,F15.0/
     &15X,'DRAFT',F23.2,F15.2/)
 3000 FORMAT (15X,'BLOCK COEFFICIENT',F12.3,F15.3/15X,'L C G (FORWARD)',
     &F12.1,F15.1/15X,'GYRADIUS',F19.1,F15.1///)
 4000 FORMAT (10X,'PROPULSION AND RUDDER CHARACTERISTICS AND STEERING',
```

```
     &' CRITERIA'///15X,'NUMBER OF SHAFTS',I14/
     &15X,'HORSEPOWER PER SHAFT',F11.0/15X,'MAXIMUM RPM',F20.0/
     &15X,'SHAFT OFFSET FROM CL',F11.0/)
5000 FORMAT (15X,'BLADES PER PROPELLER',I10/15X,'DIAMETER',F24.1/
     &15X,'PITCH',F27.1/15X,'AREA RATIO',F24.3//
     &15X,'WAKE FRACTION',F22.4/15X,'THRUST DEDUCTION',F19.4/)
6000 FORMAT (15X,'AREA PER STEERING RUDDER',F7.0/15X,
     &'AREA PER FLANKING RUDDER',F7.0/15X,'MAXIMUM RUDDER ANGLE',F14.3/
     &15X,'MAXIMUM RUDDER RATE',F15.3///)
7000 FORMAT (10X,'INITIAL BARGE/TOWBOAT VELOCITY AND ORIENTATION'///
     &15X,'BOWCLR    SPEED    GAMMA    YAW    YAW    RADIAL'/
     &42X,'ANGLE   RATE   OFFSET'/
     &15X,F6.2,F9.2,1X,3F8.4,F8.3///)
8000 FORMAT (10X,'INTEGRATION CONTROL PARAMETERS'///
     &15X,'NCUTS  FIRST-STEP   STEP-SIZE   REL-ERROR   ABS-ERROR'/
     &15X,I5,2F12.4,2F12.8)
9000 FORMAT ('1')
      RETURN
      END
      SUBROUTINE TRPOUT
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
      COMMON/CMDATA/TEMP(7),NHEAD(20),LABEL,ITAG,JTAG
      COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMTIME/ACUMT(100),STIME(100),NXTCRV,LSTCRV,NXTREV,LSTREV
      LOGICAL NXTCRV,LSTCRV,NXTREV,LSTREV
C***   THIS SUBROUTINE PRINTS A SUMMARY OF ELAPSED TIME FOR A TRIP
      CALL DATOUT
      WRITE (LPRT,1000) NHEAD,NCASE
      DO 100 I=INSEG,NSEG
      SEGD=(SDAT(1,I,1)+SDAT(1,I,2))/2.
      SEGW=SDAT(1,I,2)-SDAT(1,I,1)
      WRITE (LPRT,2000) I,SEGD,SEGW,SANG(1,I),STIME(I),ACUMT(I)
 100  CONTINUE
1000 FORMAT (///20X,20A4/////10X,'SUMMARY OF ELAPSED TIME FOR CASE',
     &I3//10X,'SEGMENT    RADIUS     WIDTH     ANGLE    ',
     &'TIME(SEG)    TIME(ACUM)'//)
2000 FORMAT (10X,I5,2X,2F10.0,F10.4,2F12.2)
      RETURN
      END
      SUBROUTINE SETY (Y,ALPHA)
C***  ***IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CMPRNT/IN,LPRT,MONITR
      COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
      COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
     &RPMHI(3),ISEG
      DIMENSION Y(6)
      SEGD=(SDAT(1,ISEG,1)+SDAT(1,ISEG,2))/2.
```

```
21500       Y(6)=SEGD+DRAD
21510       Y(2)=ALPHA
21520       Y(3)=CDOT
21530       Y(4)=HEAD
21540       Y(5)=SPEED*SIN(GAMMA)
21550       Y(1)=SPEED/Y(6)*COS(GAMMA)
21560       RETURN
21570       END
21580       SUBROUTINE SETIC (Y,TSEG,ALPHA,NTIMES,NEMORY)
21590C***   ***IMPLICIT REAL*8 (A-H,O-Z)
21600       COMMON/CMPRNT/IN,LPRT,MONITR
21610       COMMON/CMCNTR/NCARD,NCASE,NERORS,NPAGE,KPATH,KPRINT,LINECT
21620       COMMON/CMINIT/SPDIN,CURENT,GAMIN,HEADIN,CDOTIN,DRADIN,INSEG
21630       COMMON/CMSTAT/RPM(3),SPEED,DELTA(2,3),GAMMA,HEAD,CDOT,DRAD,
21640      &RPMHI(3),ISEG
21650       COMMON/CMBEGN/BEGRPM(3),BEGSPD,BEGDEL(2,3),BEGGAM,BEGHED,BEGCDT,
21660      &BEGDRD
21670       COMMON/CMSTER/BOWCLR,STRSLO,STRSLI,STRCOR,STRBK,RGAIN,RPMAX
21680       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
21690       COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,SIGN,CLBOW,BOWCL
21700       COMMON/CMTIME/ACUMT(100),STIME(100),NXTCRV,LSTCRV,NXTREV,LSTREV
21710       COMMON/SCR2/NOUTS(8),POUTS(8),NSTRS,KNT
21720       COMMON/SCR3/HISTRY(10)
21730       DIMENSION Y(6)
21740       LOGICAL IMPROV,JUMP
21750       LOGICAL NXTCRV,LSTCRV,NXTREV,LSTREV
21760       MSIZE=6
21770       CALL TIMCOR (Y,TSEG,3)
21780       NPOINT=NTIMES-(NTIMES-1)/MSIZE*MSIZE
21790       IF(DELTA(1,1).GT.1.57)GOTO40
21800       IF(DELTA(1,1).LT.-1.57)GOTO15
21810       HISTRY(1)=TSEG
21820       HISTRY(2)=ALPHA
21830       HISTRY(3)=RPM(1)
21840       HISTRY(4)=RPM(2)
21850       HISTRY(5)=RPM(3)
21860       HISTRY(6)=SPEED
21870       HISTRY(7)=GAMMA
21880       HISTRY(8)=HEAD
21890       HISTRY(9)=CDOT
21900       HISTRY(10)=DRAD
21910       GO TO 100
21920    15 JUMP=.TRUE.
21930       TSEG=0.0
21940       ALPHA=0.0
21950       SPEED=BEGSPD
21960       GAMMA=BEGGAM
21970       HEAD=BEGHED
21980       CDOT=BEGCDT
21990       DRAD=BEGDRD
```

```
22000       GO TO 50
22010    40 CONTINUE
22020       TSEG=HISTRY(1)
22030       ALPHA=HISTRY(2)
22040       RPM(1)=HISTRY(3)
22050       RPM(2)=HISTRY(4)
22060       RPM(3)=HISTRY(5)
22070       SPEED=HISTRY(6)
22080       GAMMA=HISTRY(7)
22090       HEAD=HISTRY(8)
22100       CDOT=HISTRY(9)
22110       DRAD=HISTRY(10)
22120    50 CONTINUE
22130       CALL SETY (Y,ALPHA)
22140C***   CALL CHANCK (Y(6),Y(4),Y(2),ISEG)
22150   100 CONTINUE
22160       RETURN
22170       END
22180       SUBROUTINE POLY3 (XVAL,YVAL,COEF)
22190C***   ***IMPLICIT REAL*8 (A-H,O-Z)
22200       COMMON/CMPRNT/IN,LPRT,MONITR
22210       DIMENSION X(6),XY(6),XVAL(3),YVAL(3),COEF(3)
22220       EQUIVALENCE (X(1),X1),(X(2),X2),(X(3),X3)
22230       EQUIVALENCE (X(4),X4),(X(5),X5),(X(6),X6)
22240       EQUIVALENCE (XY(1),X1Y),(XY(2),X2Y),(XY(3),X3Y)
22250       DO 10 I=1,6
22260       X(I)=0.0
22270       XY(I)=0.0
22280    10 CONTINUE
22290       DO 30 I=2,6
22300       DO 20 J=1,3
22310       X(I)=X(I)+XVAL(J)**I
22320    20 CONTINUE
22330    30 CONTINUE
22340       DO 50 I=1,3
22350       DO 40 J=1,3
22360       XY(I)=XY(I)+YVAL(J)*XVAL(J)**I
22370    40 CONTINUE
22380    50 CONTINUE
22390       D=X2*(X4*X6-X5*X5)-X3*(X3*X6-X4*X5)+X4*(X3*X5-X4*X4)
22400       IF (D .EQ. 0.0) WRITE (LPRT,1000)
22410 1000 FORMAT ('DETERMINANT IS ZERO IN POLY3')
22420       COEF(1)=(X1Y*(X4*X6-X5*X5)-X2Y*(X3*X6-X4*X5)+X3Y*(X3*X5-X4*X4))/D
22430       COEF(2)=(-X1Y*(X3*X6-X4*X5)+X2Y*(X2*X6-X4*X4)-X3Y*(X2*X5-X3*X4))/D
22440       COEF(3)=(X1Y*(X3*X5-X4*X4)-X2Y*(X2*X5-X3*X4)+X3Y*(X2*X4-X3*X3))/D
22450       RETURN
22460       END
22470       SUBROUTINE SELECT (PARAM,VALUE,NEND,LIM)
22480C***   ***IMPLICIT REAL*8 (A-H,O-Z)
22490       COMMON/CMPRNT/IN,LPRT,MONITR
```

```
      DIMENSION PARAM(7),LIM(3)
      LIMIN=1
      LIMAX=NEND
C***   CHECK *PARTIM(I,J,K)* FOR ((-1)), AND ADJUST LIMIN AND LIMAX
      PMIN=PARAM(LIMIN)
      PMAX=PARAM(LIMAX)
      IF (PMIN .LE. VALUE  .AND.  PMAX .GT. VALUE) GO TO 5
      WRITE (LPRT,1000) VALUE,PMIN,PMAX
      GO TO 100
    5 CONTINUE
      DO 20 I=LIMIN,LIMAX
      IBACK=LIMAX+1-I
      IF (PARAM(I) .GT. VALUE) GO TO 10
      PLO=PARAM(I)
      MINN=I
   10 CONTINUE
      IF (PARAM(IBACK) .LE. VALUE) GO TO 20
      PHI=PARAM(IBACK)
      MAXX=IBACK
   20 CONTINUE
      IF (MAXX .EQ. MINN+1) GO TO 25
      WRITE (LPRT,2000) MINN,PARAM(MINN),MAXX,PARAM(MAXX)
   25 CONTINUE
      IF (MINN .GT. 1) GO TO 30
      NXTRA=2
      MAXX=3
      GO TO 100
   30 CONTINUE
      IF (MAXX .LT. LIMAX) GO TO 40
      NXTRA=LIMAX-1
      MINN=LIMAX-2
      GO TO 100
   40 CONTINUE
      AVG=(PARAM(MINN)+PARAM(MAXX))/2.0
      IF (AVG .GT. VALUE) GO TO 50
      NXTRA=MAXX
      MAXX=MAXX+1
      GO TO 100
   50 CONTINUE
      NXTRA=MINN
      MINN=MINN-1
  100 CONTINUE
      LIM(1)=MINN
      LIM(2)=NXTRA
      LIM(3)=MAXX
 1000 FORMAT ('   VALUE',F15.5,'  IS OUT OF RANGE',2F15.5)
 2000 FORMAT ('   PARAMETERS OUT OF SEQUENCE',2(I5,2X,F15.5))
      RETURN
      END
      SUBROUTINE HYDRO (ANGRAT,SPD,APBETA,SMOM,FSWAY)
```

```
23000C***   ***IMPLICIT REAL*8 (A-H,O-Z)
23010       COMMON/CMPRNT/IN,LPRT,MONITR
23020       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
23030       COMMON/CMCNST/RHO,GRAV,PI
23040       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
23050       COMMON/CMCOEF/A(10),B(10),AA(10),BB(10),COEF(5)
23060       COMMON/CMCLOG/NPOLY,NSTRIP,KORPOL,KORSTR
23070       TLEN4=TLEN*TLEN3
23080       HRO=RHO/2.0
23090       V=-SPD*SIN(APBETA)
23100       T1=HRO*TLEN2*SPD*SPD
23110       T2=HRO*TLEN2*SPD*V
23120       T3=HRO*TLEN3*SPD*ANGRAT
23130       T5=HRO*TLEN3/SPD*ANGRAT*V*V
23140       T6=HRO*TLEN4/SPD*ANGRAT*ANGRAT*V
23150       T7=HRO*TLEN2/SPD*V*V*V
23160       SMOM=A(1)*T1+A(2)*T2+A(3)*T3+A(5)*T5+A(6)*T6+A(7)*T7
23170       SMOM=SMOM*TLEN
23180       FSWAY=B(1)*T1+B(2)*T2+B(3)*T3+B(5)*T5+B(6)*T6+B(7)*T7
23190       RETURN
23200       END
23210       SUBROUTINE HYCOEF
23220C***   ***IMPLICIT REAL*8 (A-H,O-Z)
23230       COMMON/CMPRNT/IN,LPRT,MONITR
23240       COMMON/CMWIDE/TOWWID,BOTWID,NBARW
23250       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
23260       COMMON/CMCHAR/TOWDRF,BOTDRF,TOWBC,BOTBC
23270       COMMON/CMCNST/RHO,GRAV,PI
23280       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
23290       COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
23300       COMMON/CMCOEF/A(10),B(10),AA(10),BB(10),COEF(5)
23310       COMMON/CMCLOG/NPOLY,NSTRIP,KORPOL,KORSTR
23320       LOGICAL NPOLY,NSTRIP,KORPOL,KORSTR
23330       TLEN=TOWLEN+BOTLEN
23340       TLEN2=TLEN*TLEN
23350       TLEN3=TLEN*TLEN2
23360       TLEN4=TLEN*TLEN3
23370       TLEN5=TLEN*TLEN4
23380       IF (NPOLY) GO TO 800
23390       IF (NSTRIP) GO TO 100
23400       IF (KORPOL) GO TO 900
23410       IF (KORSTR) GO TO 200
23420       CDT=1.5
23430       CDB=0.5
23440       CAT=0.45
23450       CAB=0.45
23460       CAS=0.0
23470       GO TO 300
23480   100 CONTINUE
23490       KORPOL=.FALSE.
```

```
23500  200 CONTINUE
23510      CDT=COEF(1)
23520      CDB=COEF(2)
23530      CAT=COEF(3)
23540      CAB=COEF(4)
23550      CAS=COEF(5)
23560  300 CONTINUE
23570      ST=CAT*TOWBC*TOWWID*TOWDRF
23580      SB=CAB*BOTBC*BOTWID*BOTDRF
23590      DB=CGAFT+BOTLEN/2.0
23600      DT=CGFWD-TOWLEN/2.0
23610      TT=TOWDRF*TOWLEN
23620      BO=BOTDRF*BOTLEN
23630      CC=(CGAFT*CGAFT*CGAFT+CGFWD*CGFWD*CGFWD)/3.0
23640      CBC=(-CGAFT*CGAFT*CGAFT+(CGAFT+BOTLEN)**3)/3.0
23650      B(2)=-(CDT*TT+CDB*BO)/TLEN2
23660      A(3)=-(CDT*TOWDRF*CC+CDB*BOTDRF*CBC)/TLEN4
23670      A(2)=(CDB*BO*DB-CDT*TT*DT)/TLEN3
23680      B(3)=A(2)
23690      A(9)=-(ST*CC+SB*CBC)/TLEN5
23700      B(10)=-(ST*TOWLEN+SB*BOTLEN)/TLEN3
23710      GO TO 1000
23720  800 CONTINUE
23730      KORSTR=.FALSE.
23740  900 CONTINUE
23750      DO 950 K=1,10
23760      A(K)=AA(K)
23770      B(K)=BB(K)
23780  950 CONTINUE
23790 1000 CONTINUE
23800      RETURN
23810      END
23820      SUBROUTINE BOWTHR (VADV,BTPUSH,BTSWAY,BTMOM)
23830C***  ***IMPLICIT REAL*8 (A-H,O-Z)
23840      COMMON/CMPRNT/IN,LPRT,MONITR
23850      COMMON/CMSIZE/CGAFT,CGFWD,EFLEN,EFDRF,EFBEAM,TONS
23860      COMMON/CMCLER/CLIN,XLIN,CLOUT,XLOUT,SIGN,CLBOW,BOWCL
23870      COMMON/CMBOWT/BTHRUS(7),BTSPD(7),BTMAX,BTPOS,BTGAIN,NBTSPD,IDBT
23880      LOGICAL IDBT
23890      DIMENSION BTCOEF(3),BTH(3),V(3),LIMB(3)
23900      IF (IDBT) GO TO 5
23910      BTPUSH=0.0
23920      BTSWAY=0.0
23930      BTMOM=0.0
23940      GO TO 1000
23950    5 CONTINUE
23960      CALL SELECT (BTSPD,VADV,NBTSPD,LIMB)
23970      DO 10 I=1,3
23980      V(I)=BTSPD(LIMB(I))
23990      BTH(I)=BTMAX-BTHRUS(LIMB(I))
```

```
24000    10 CONTINUE
24010       CALL POLY3 (V,BTH,BTCOEF)
24020       BTFORC=0.0
24030       DO 20 K=1,3
24040       BTFORC=BTFORC+BTCOEF(K)*VADV**K
24050    20 CONTINUE
24060       BTFORC=BTMAX-BTFORC
24070       IF (BTFORC .GT. 0.0) GO TO 30
24080       BTPUSH=BTFORC
24090       BTSWAY=0.0
24100       BTMOM=0.0
24110       GO TO 1000
24120    30 CONTINUE
24130       BTLIM=BTMAX*BTGAIN*(BOWCL-CLBOW)*SIGN/BOWCL
24140       IF (BTLIM .GT. BTMAX) BTLIM=BTMAX
24150       IF (BTLIM .LT. -BTMAX) BTLIM=-BTMAX
24160       IF (ABS(BTLIM) .LT. BTFORC) GO TO 100
24170       BTPUSH=0.0
24180       BTSWAY=BTFORC*BTLIM/ABS(BTLIM)
24190       BTMOM=BTSWAY*(CGFWD+BTPOS)
24200       GO TO 1000
24210   100 CONTINUE
24220       BTSWAY=BTLIM
24230       BTMOM=BTSWAY*(CGFWD+BTPOS)
24240       BTPUSH=SQRT(BTFORC*BTFORC-BTSWAY*BTSWAY)
24250  1000 CONTINUE
24260       RETURN
24270       END
24280       SUBROUTINE DECEL (FRMSPD,TOSPD,TDIF)
24290C***   ***IMPLICIT REAL*8 (A-H,O-Z)
24300       COMMON/CMNTIG/FIRSTP,STEP,EPS,AB,NCUTS
24310       COMMON/CMDISP/TOWDSP,BOTDSP,GYRAD,TMASS,ZNERTA,TLEN,TLEN2,TLEN3
24320       TDIF=0.0
24330       DSLOW=0.0
24340       NTRY=0
24350       BADS=FRMSPD
24360       IF (FRMSPD .LE. TOSPD) GO TO 999
24370    10 CONTINUE
24380       CALL RESIST (BADS,DRAG)
24390       NTRY=NTRY+1
24400       BADT=BADS-DRAG*FIRSTP/TMASS
24410       DSLOW=DSLOW+FIRSTP*(BADS+BADT)/2.0
24420       BADS=BADT
24430       IF (BADS .GT. TOSPD .AND. NTRY .LT. 200) GO TO 10
24440       TSLOW=FIRSTP*NTRY
24450       TDIF=TSLOW-DSLOW/FRMSPD
24460   999 RETURN
24470       END
24480       SUBROUTINE BARDIS(ALPHA,RADIN,PSI,XPOS,DPERI,DPERO,
24490      &PERRAD,PERTAN,ISSEG0,ISEG0)
```

```
24500       COMMON/CMSTAT/DUM(17),ISEGIN
24510       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
24520       DIMENSION DISTI(10),DISTO(10)
24530       RADIUS=RADIN
24540       ISEG=ISEGIN
24550       DISTI(1)=9.999E+3
24560       DISTO(1)=9.999E+3
24570       PI180=3.1415927/180.
24580       RAD=ALPHA
24590C      CALCULATE BARGE POSITION IN SEGMENT "ISEGIN"
24600       XGBAR=RADIUS*COS(RAD)
24610       YGBAR=RADIUS*SIN(RAD)
24620C       CALCULATE POINT POSITION IN SEGMENT "ISEGIN"
24630       RAD=(1.5708-PSI+ALPHA)
24640       IF(ABS(XPOS).LE.0.001)XPOS=0.001*SIGN(1.0,XPOS)
24650       XP=XGBAR+XPOS*COS(RAD)
24660       YP=YGBAR+XPOS*SIN(RAD)
24670C      CALCULATE ANGLE OF POINT WITHIN SEGMENT
24680       ANGPT=ATAN2(YP,XP)/PI180
24690       ANGSEG=SANG(1,ISEG)
24700C      CHECK TO SIGN OF SEGMENT ANGLE
24710       IF(ANGSEG) 10,20,30
24720    10 ANG1=A0N2PI(ANGSEG)
24730       ANG2=360.
24740       ANGPT0=A0N2PI(ANGPT)
24750       GO TO 40
24760    20 RETURN
24770    30 ANG1=0.0
24780       ANG2=A0N2PI(ANGSEG)
24790       ANGPT0=ANGPT
24800C       CHECK IF POINT IS WITHIN SEGMENT "ISEGIN"
24810    40 ISADD=0
24820       IF(ANGPT0-ANG1) 50,80,60
24830    50 ISADD=-1
24840       GO TO 80
24850    60 IF(ANGPT0-ANG2) 80,80,70
24860    70 ISADD=1
24870    80 CONTINUE
24880C      IF BARGE POINT IS NOT IN "ISEGIN" DEFINE POINT AND BARGE
24890C      CG IN NEW SEGMENT "ISEG+ISADD"
24900       IF(ISADD) 90,100,90
24910    90 CALL COORD(XP,YP,ISEG,ISADD,XPI,YPI)
24920       CALL COORD(XGBAR,YGBAR,ISEG,ISADD,XG,YG)
24930       XP=XPI
24940       YP=YPI
24950C***   WRITE(5,982)XP,YP
24960C***   982 FORMAT('  XP=',F8.2,'  YP=',F8.2)
24970       XGBAR=XG
24980       YGBAR=YG
24990       ISEG=ISEG+ISADD
```

```
25000       ANGPT=ATAN2(YP,XP)/PI180
25010       ANGPT0=A0N2PI(ANGPT)
25020       RADIUS=SQRT(XP*XP+YP*YP)
25030   100 CONTINUE
25040C      FIND SUB-SEGMENT WITHIN "ISEG" THAT POINT IS LOCATED
25050       ANG=0.0
25060       NSSEG=NANG(ISEG)+1
25070       SN=SIGN(1.,SANG(1,ISEG))
25080       ANGPT=ANGPT0+180.*(SN-1.)
25090       DO 110 ISS=2,NSSEG
25100       ISSEG0=ISS
25110       PERTAN=(ANGPT-ANG)/SANG(ISS,ISEG)
25120       ANG=ANG+SANG(ISS,ISEG)
25130       IF(SN*(ANGPT-ANG)) 120,120,110
25140   110 CONTINUE
25150   120 CONTINUE
25160C      CALCULATE  INSIDE AND OUTSIDE PERPENDICULAR DISTANCES
25170       I1=MAX0(ISEG-1,1)
25180       I2=MIN0(ISEG+1,NSEG)
25190       IO=0
25200       II=0
25210       DO 190 IS = I1,I2
25220       SN=SIGN(1.,SANG(1,IS))
25230       ISADD=IS-ISEG
25240       CALL COORD(XP,YP,ISEG,ISADD,X,Y)
25250       CALL COORD(XGBAR,YGBAR,ISEG,ISADD,XG,YG)
25260       DY=XG-X
25270       DX=Y-YG
25280       NSSEG=NANG(IS)+1
25290       X2I = SDAT(1,IS,1)
25300       Y2I = 0.0
25310       X2O = SDAT(1,IS,2)
25320       Y2O = 0.0
25330       ANG=0.0
25340       DO 190 ISS = 2 , NSSEG
25350       X1I = X2I
25360       Y1I = Y2I
25370       X1O = X2O
25380       Y1O = Y2O
25390       ANG = ANG + SANG(ISS,IS)
25400       R2I = SDAT(ISS,IS,1)
25410       R2O = SDAT(ISS,IS,2)
25420       RAD = ANG*PI180
25430       COR = COS(RAD)
25440       SIR = SIN(RAD)
25450       X2I = R2I*COR
25460       Y2I = R2I*SIR
25470       X2O = R2O*COR
25480       Y2O = R2O*SIR
25490C       INSIDE INTERSECTION CALCULATIONS
```

```
25500       CALL LININT(X1I,Y1I,X2I,Y2I,X,Y,DX,DY,DIST,ISTAT)
25510       IF(ISTAT) 140,140,130
25520   130 II=II+1
25530       DISTI(II)=SN*DIST
25540C       OUTSIDE INTERSECTION CALCULATIONS
25550   140 CALL LININT(X1O,Y1O,X2O,Y2O,X,Y,DX,DY,DIST,ISTAT)
25560       IF(ISTAT) 160,160,150
25570   150 IO = IO + 1
25580       DISTO(IO)=-SN*DIST
25590   160 CONTINUE
25600       IF(ISADD) 190,170,190
25610   170 IF(ISS-ISSEG0) 190,180,190
25620   180 CONTINUE
25630       CALL LININT(X1I,Y1I,X2I,Y2I,X,Y,X,Y,DIN,ISTAT)
25640       CALL LININT(X1O,Y1O,X2O,Y2O,X,Y,X,Y,DOUT,ISTAT)
25650       PERRAD=DIN/(DIN-DOUT)
25660   190 CONTINUE
25670       ISEG0=ISEG
25680       DPERI=DISTI(1)
25690       DPERO=DISTO(1)
25700       IO=MAX0(IO,1)
25710       II=MAX0(II,1)
25720       IF(IO.EQ.1)GOTO 201
25730       DO 200 I=2,IO
25740   200 IF(ABS(DISTO(I)).LT.ABS(DPERO))DPERO=DISTO(I)
25750   201 IF(II.EQ.1)GOTO 203
25760       DO 202 I=2,II
25770   202 IF(ABS(DISTI(I)).LT.ABS(DPERI))DPERI=DISTI(I)
25780   203 CONTINUE
25790       RETURN
25800       END
25810       SUBROUTINE COORD(XP,YP,ISEG,ISADD,XPI,YPI)
25820C      SUBROUTINE CHANGES COORDINATE SYSTEM OF POINT (XP,YP) IN SEGMENT
25830C      "ISEG" TO (XPI,YPI) IS SEGMENT "ISEG+ISADD"
25840       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
25850C       CHECK WHICH SEGMENT FOR POINT
25860       PI180=3.1415927/180.
25870       IF(ISADD) 300,200,100
25880C      CALCULATE POINT COORD IN SEGMENT FORWARD OF BARGE
25890   100 AL1=SANG(1,ISEG)
25900       NNS=NANG(ISEG)+1
25910       RAD1=AL1
25920       IF(AL1*SANG(1,ISEG+1)) 110,500,120
25930   110 DR=SDAT(NNS,ISEG,1)+SDAT(1,ISEG+1,2)
25940       ROT=180.+AL1
25950       GO TO 400
25960   120 DR=SDAT(NNS,ISEG,1)-SDAT(1,ISEG+1,1)
25970       ROT=AL1
25980       GO TO 400
25990C      LEAVE POINT COORDS IN PRESENT SEGMENT
```

```
26000   200 XPI=XP
26010       YPI=YP
26020       RETURN
26030C       CALCULATE POINT COORDS IN SEGMENT AFT OF BARGE
26040   300 AL1=SANG(1,ISEG-1)
26050       NNS=NANG(ISEG-1)+1
26060       RAD1=0.0
26070       IF(AL1*SANG(1,ISEG)) 310,500,320
26080   310 DR=SDAT(NNS,ISEG-1,1)+SDAT(1,ISEG,2)
26090       ROT=180.-AL1
26100       GO TO 400
26110   320 DR=SDAT(1,ISEG,1)-SDAT(NNS,ISEG-1,1)
26120       ROT=-AL1
26130   400 RAD=RAD1*PI180
26140       DH=DR*COS(RAD)
26150       DK=DR*SIN(RAD)
26160       XPT=XP-DH
26170       YPT=YP-DK
26180       RAD=ROT*PI180
26190       COR=COS(RAD)
26200       SIR=SIN(RAD)
26210       XPI=XPT*COR+YPT*SIR
26220       YPI=YPT*COR-XPT*SIR
26230   500 CONTINUE
26240       RETURN
26250       END
26260       FUNCTION A0N2PI(ANGIN)
26270       ANG=ANGIN
26280       IF(ANG) 1,2,3
26290     1 ANG=ANG+360.
26300       IF(ANG)1,2,2
26310     4 ANG=ANG-360.
26320     3 IF(ANG-360.) 2,4,4
26330     2 A0N2PI=ANG
26340       RETURN
26350       END
26360       SUBROUTINE LININT(X1,Y1,X2,Y2,X,Y,DX,DY,DIST,ISTAT)
26370       ISTAT=-1
26380       DIST=0.0
26390       A1=Y2-Y1
26400       B1=X1-X2
26410       C1=X1*A1+Y1*B1
26420       A2=DY
26430       B2=-DX
26440       C2=X*A2+Y*B2
26450       RAD=A2*B1-A1*B2
26460       IF(RAD.EQ.0.) RETURN
26470       XT=(B1*C2-B2*C1)/RAD
26480       YT=-(A1*C2-A2*C1)/RAD
26490       YMAX=AMAX1(Y1,Y2)
```

```
26500       YMIN=AMIN1(Y1,Y2)
26510       XMAX=AMAX1(X1,X2)
26520       XMIN=AMIN1(X1,X2)
26530       IF(XT-XMIN+0.00001) 60,10,10
26540    10 IF(XT-XMAX-0.00001) 20,20,60
26550    20 IF(YT-YMIN+0.00001) 60,30,30
26560    30 IF(YT-YMAX-0.00001) 40,40,60
26570    40 DIST=SQRT((XT-X)**2+(YT-Y)**2)
26580       ISTAT=1
26590       DIFF=-C1+A1*X+B1*Y
26600       DIST=SIGN(1.,DIFF)*DIST
26610    60 CONTINUE
26620       RETURN
26630       END
26640       SUBROUTINE CURRNT(ALPHA,PSI,RADIUS,CTAN,CRAD,CROT)
26650       COMMON/CMLONG/TOWLEN,BOTLEN,DUM2(4),NBARL
26660       COMMON/CMROUT/SDAT(11,10,2),CUR(11,10,3),SANG(11,10),NANG(10),NSEG
26670       COMMON/CMSTAT/DUM3(17),ISEG
26680       COMMON/CMDISP/DUM(5),TLEN,TLEN2,TLEN3
26690       COMMON/CMSIZE/CGAFT,CGFWD,DUM1(4)
26700       DIMENSIONXPOSN(3),CTI(3),CTO(3),CTA(3),CRA(3)
26710       XPOSN(1)=0.0
26720       XPOSN(2)=CGFWD
26730       XPOSN(3)=-CGAFT-BOTLEN
26740C***   WRITE(5,875)ALPHA,RADIUS,PSI
26750   875 FORMAT(/' ALPHA',F7.3,'   RADIUS',F7.0,'   PSI',F7.3)
26760       DO 10 N=1,3
26770       CALL BARDIS(ALPHA,RADIUS,PSI,XPOSN(N),D1,D2,PRAD,PTAN,
26780      &ISUBO,ISEGO)
26790       ISUBM=ISUBO-1
26800       CTI(N)=CUR(ISUBO,ISEGO,1)*PTAN+CUR(ISUBM,ISEGO,1)*(1.-PTAN)
26810       CTO(N)=CUR(ISUBO,ISEGO,2)*PTAN+CUR(ISUBM,ISEGO,2)*(1.-PTAN)
26820       CTA(N)=CTO(N)*PRAD+CTI(N)*(1.-PRAD)
26830       CRA(N)=CUR(ISUBO,ISEGO,3)*PTAN+CUR(ISUBM,ISEGO,3)*(1.-PTAN)
26840       IF(ISEGO.EQ.ISEG)GOTO10
26850       CHSN=SIGN(1.0,SANG(1,ISEG))*SIGN(1.0,SANG(1,ISEGO))
26860       CRA(N)=CRA(N)*CHSN
26870C***   WRITE(5,876)CTO(N),CTI(N),CTA(N),CRA(N)
26880   876 FORMAT(' CTO',F5.1,'  CTI',F5.1,'  CTAN',F5.2,'  CRAN',F5.2)
26890    10 CONTINUE
26900       CTAN=CTA(1)
26910       CRAD=CRA(1)
26920       CLATB=CRA(2)*COS(PSI)+CTA(2)*SIN(PSI)
26930       CLATS=CRA(3)*COS(PSI)+CTA(3)*SIN(PSI)
26940       CROT=(CLATB-CLATS)/TLEN
26950C***   WRITE(5,877)CTAN,CRAD,CROT
26960   877 FORMAT(' CURTAN',F5.2,'   CURRAD',F5.2,'   CURROT',F5.2)
26970       RETURN
26980       END
26990       SUBROUTINE WINDFO(ALPHA,PSI,FWLON,FWLAT,FWROT)
```

```
27000       COMMON/CMWIDE/TOWWID,BOTWID,NBARW
27010       COMMON/CMLONG/TOWLEN,BOTLEN,CGTOW,CGBOT,TOWK,BOTK,NBARL
27020       COMMON/CMCHAR/TOWDRF,BOTDRF,TOWBC,BOTBC
27030       COMMON/CMSTAT/DUM3(17),ISEG
27040       COMMON/CMCNST/RHO,GRAV,PI
27050       COMMON/CMWIND/SWIND(10),DWIND(10)
27060       AX=(20.-TOWDRF)*TOWWID
27070       AY=(15.-TOWDRF)*(TOWLEN+BOTLEN)
27080       RHOA=RHO/800./2.
27090       ALREL=DWIND(ISEG)*PI/180.-ALPHA
27100       WSR=SWIND(ISEG)*COS(ALREL)
27110       WST=SWIND(ISEG)*SIN(ALREL)
27120       WX=WSR*SIN(PSI)+WST*COS(PSI)
27130       WY=WSR*COS(PSI)-WST*SIN(PSI)
27140       CDX=1.0*RHOA*AX
27150       CDY=1.0*RHOA*AY
27160       CMX=0.0*RHOA*AX
27170       CMY=0.0*RHOA*AY
27180       WX=WX*ABS(WX)
27190       WY=WY*ABS(WY)
27200       FWLON=CDX*WX
27210       FWLAT=CDY*WY
27220       FWROT=CMX*WX+CMY*WY
27230       RETURN
27240       END
```